Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 20 (2006)

Quantum Hardcore Functions by Complexity-Theoretical Quantum List Decoding*

Akinori Kawachi

Graduate School of Information Science and Engineering, Tokyo Institute of Technology
2-12-1 Ookayama, Meguro-ku, Tokyo 152-8552, Japan

Tomoyuki Yamakami

ERATO-SORST Quantum Computation and Information Project, Japan Science and Technology Agency
5-28-3 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

Abstract

Hardcore functions have been used as a technical tool to construct secure cryptographic systems; however, little is known on their quantum counterpart, called *quantum hardcore functions*. With a new insight into fundamental properties of quantum hardcores, we present three new quantum hardcore functions for any (strong) quantum one-way function. We also give a "quantum" solution to Damgård's question (CRYPTO'88) on a classical hardcore property of his pseudorandom generator, by proving its quantum hardcore property. Our major technical tool is the new notion of quantum list-decoding of "classical" error-correcting codes (rather than "quantum" error-correcting codes), which is defined on the platform of computational complexity theory and computational cryptography (rather than information theory). In particular, we give a simple but powerful criterion that makes a polynomial-time computable classical block code (seen as a function) a quantum hardcore for all quantum one-way functions. On their own interest, we construct elegant quantum list-decoding algorithms for classical block codes whose associated quantum states (called codeword states) form a nearly phase orthogonal basis. In particular, we prove that circulant codes that enjoy a multiplicative property are quantum list-decodable.

keywords: quantum hardcore, quantum one-way, quantum list-decoding, codeword state, phase orthogonal, presence, Johnson bound
AMS Subject Classifications: 14G50, 81P68, 94A60

1 From Hardcores to List Decoding

Modern cryptography heavily relies on computational hardness and pseudorandomness. One of its key notions is a *hardcore bit* for a one-way function—a bit that could be determined from all the information available to the mighty adversary but still looks random to any "feasible" adversary. A hardcore function transforms the onewayness into pseudorandomness by generating such hardcore bits of a given one-way function. Such a hardcore function is a crucial element of a construction of a pseudorandom generator as well as a bit commitment protocol from a one-way permutation. A typical example is the inner product mod two function $\mathrm{GL}_x(r)$ of Goldreich and Levin [15], computing the bitwise inner product modulo two, $\langle x, r\rangle$, which constitutes a hardcore bit for any (strong) one-way function.† Since $\mathrm{GL}_x(r)$ equals the rth bit of the codeword $\mathrm{HAD}_x^{(2)} = (\langle x, 0^n\rangle, \langle x, 0^{n-1}1\rangle, \cdots, \langle x, 1^n\rangle)$ of message x of a binary Hadamard code, Goldreich and Levin essentially gave a polynomial-time list-decoding algorithm for this Hadamard code. In the recent literature, list-decoding has kept playing a key role in a general construction of hardcores [2, 21].

Thirteen years later, the "quantum" hardcore property (i.e., a hardcore property against any feasible "quantum" adversary) of $\mathrm{GL}_x(\cdot)$ was shown by Adcock and Cleve [1], who implicitly gave a simple and efficient quantum algorithm that recovers x from the binary Hadamard code by exploiting the robust nature of a quantum algorithm of Bernstein and Vazirani [6]. The simplicity of the proof of Adcock and Cleve can be best compared to the original proof of Goldreich and Levin, who employed a rather complicated algorithm with powerful techniques: self-correction property of the aforementioned Hadamard code and

*An extended abstract appeared in the Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP 2006), Lecture Notes in Computer Science, Vol.4052 (Part II), pp.216–227. Venice, Italy. July 10–14, 2006.

†Literally speaking, this statement is slightly misleading. To be more accurate, such a hard-core function concerns only the one-way function of the form $f'(x, r) = (f(x), r)$ with $|r| = poly(|x|)$ induced from an arbitrary strong one-way function f. See, e.g., [14] for a detailed discussion.

ISSN 1433-8092

pairwise independent sampling. This highlights a significant role of robust quantum computation in list-decoding (and thus hardcores); however, it has been vastly unexplored until our work except for a quantum decoder of Barg and Zhou [5] for simplex codes. No other quantum hardcore has been proven so far. The efficiency of robust quantum algorithms with access to biased oracles has been also discussed in a different context [3, 8, 23].

As our main result, we present three new quantum hardcore functions: *Hadamard codes* $\mathrm{HAD}^{(q)}$, *shifted Legendre symbol codes* SLS^p, and *pairwise equality codes* PEQ (see Section 7 for their definitions), for any (strongly) quantum one-way function. The latter two of them are not yet known to be classical hardcores (see, e.g., [17]). The quantum hardcore proof of SLS^p, in particular, uses a property of circulant matrices.

Our argument proceeds as follows. Suppose that a target function h is not any quantum hardcore for a certain quantum one-way function f' of the form $f'(x,r) = (f(x),r)$. We reduce proving the quantum hardcore property to solving the QLDP. We then reduce constructing a quantum list-decoder to constructing a quantum codeword-state decoder. Using this decoder, we can construct a polynomial-time a quantum algorithm that inverts f'. This contradicts the one-wayness of f' and hence proves the quantum hardcore property of h, as requested.

In particular, we prove the quantum hardcore property of Damgård's pseudorandom generator [10]. This gives a "quantum" solution to his question of whether his generator has the classical hardcore property (this is also listed as an open problem in [17]). Our proof technique exploits the quantum list-decodability of classical error-correcting codes (rather than quantum error-correcting codes). For our purpose, we formulate the notion of *complexity-theoretical* quantum list-decoding as a message recovery with quantum-computational errors rather than information-theoretical errors which are usually associated with transmission errors. This notion naturally expands the classical framework of list-decoding. Our goal is to present fast quantum list-decoding algorithms for the aforementioned codes.

Proving the quantum hardcore property of a given code C (seen as a function) corresponds to solving the *quantum list-decoding problem* (QLDP) for C via direct access to a *quantum-computationally (or quantumly) corrupted codeword*, which is given as a black-box oracle. The task of a quantum list-decoder is simply to list all message candidates whose codewords match the quantumly-corrupted codeword within a certain error rate bound.

One of the key concepts is *quantumly corrupted codewords*, which express the behaviors of (possibly) faulty quantum encoders. In classical list-decoding, a "classically" corrupted codeword is generated by a faulty channel as a result of its transmission error. Our scenario arises naturally if we treat transmission error as a faulty encoding process of messages to codewords. Particularly, it is useful to treat in this way when we seek applications of list-decoding in computational complexity. Another key notion of this paper is a useful quantum state, called a *(k-shuffled) codeword state*, which uses quantum "phase" to store the information on a given codeword. Similar states have appeared to play a key role in several quantum algorithms in the recent literature [6, 12, 18, 28]. In our key lemmas, we show (i) how to generate such a codeword state from *any* (even adversarial) quantumly corrupted codeword and (ii) how to convert a *codeword-state decoder* (i.e., a quantum algorithm that recovers a message x from a codeword state which is given as an input) to a quantum list-decoding algorithm working with a quantumly corrupted codeword. The robust construction made in the course of our proofs also provides a useful means, known as "hardness" reduction, which is often crucial in the security proof of a quantum cryptosystem. Because our purpose is to prove quantum hardcores, we need to discuss only codes whose codeword size is roughly an exponential in the size of messages. For such codes, using pretty good measurement [13, 20], we can present a generic way of proving their quantum list-decodability if the set of corresponding codeword states forms a "nearly" phase-orthogonal basis. This construction method is general but non-constructive. In certain cases, we can give explicitly a quantum list-decoding algorithm. An important example is nearly phase-orthogonal *circulant* codes that enjoys a certain multiplicative property. The design of our quantum list-decoder for these codes elaborates a quantum algorithm of van Dam, Hallgren, and Ip [28].

Classical list-decodable codes have provided numerous applications in classical computational complexity theory, including proving hardcores for any one-way function, hardness amplification, and derandomization (see, e.g., [27]). Because our formulation of quantum list-decoding naturally extends classical one, many classical list-decoding algorithms work in our quantum setting as well. This will make our quantum list-decoding a powerful tool in quantum complexity theory and quantum computational cryptography.

2 Quantum Hardcore Functions

We briefly give the formal definitions to the core concepts of this paper—quantum one-way functions and quantum hardcore functions. We assume the reader's basic knowledge on quantum computation. Our underlying computation model is quantum Turing machines [6, 29] and quantum circuits [30]. Informally, we use the term "quantum algorithm" to describe a description of a certain unitary operator, possibly together with a specific projection measurement at the end of a computation. For convenience, the notation $\mathcal{A}(x)$ for a quantum algorithm $\mathcal{A}$ and an input x denotes a random variable representing the outcome of the execution of $\mathcal{A}$ on input x.

We begin with the notion of *quantum one-way functions*, which straightforwardly expands the classical one-way functions introduced first by Diffie and Hellman [11] in 1976. Let $\mathbb{N}$ denote the set of all nonnegative integers.

Definition 2.1 (quantum one-wayness) A function f from $\{0,1\}^*$ to $\{0,1\}^*$ is called *(strongly) quantum one-way* if (i) there exists a polynomial-time *deterministic* algorithm G computing f and (ii) for any polynomial-time quantum algorithm $\mathcal{A}$, for any positive polynomial p, and for any sufficiently large numbers $n \in \mathbb{N}$,

$$\mathrm{Prob}_{x\in\{0,1\}^n,\mathcal{A}}\left[f(\mathcal{A}(f(x),1^n)) = f(x)\right] < \frac{1}{p(n)},$$

where x is uniformly distributed over $\{0,1\}^n$ and the subscript $\mathcal{A}$ is a random variable determined by measuring the final state of $\mathcal{A}$ in the computational basis. We consider only length-regular one-way functions, where a function f mapping $\mathbb{N}$ to $\mathbb{N}$ is called *length regular* if, for every $x \in \{0,1\}^*$, $|f(x)| = l(|x|)$ for a certain length function $l(n)$.

Because of the deterministic feature of f, all quantum one-way functions are classically one-way. For any quantum one-way function f, the notation f' denotes the function induced from f by the following scheme: $f'(x,r) = (f(x),r)$ for all $x, r \in \{0,1\}^*$ with $|r| = poly(|x|)$, where the notation (y,r) means the concatenation of y and r following y. Notice that f' is also a quantum one-way function. Throughout this paper, we deal only with quantum one-way functions of this form, which is in direct connection to the following notion of quantum hardcore functions.

The notion of a classical hardcore was first discussed by Blum and Micali [7] in 1984. A hardcore measures the hardness of predicting the value $h(x)$ from $f(x)$ without knowing x as an explicit input. A hardcore function h mapping $\{0,1\}^n$ to $\{0,1\}^{l(n)}$ is usually defined by the notion of *indistinguishability* between $h(x)$ and a truly random variable z over $\{0,1\}^{\ell(n)}$. However, a hardcore predicate (namely, a hardcore function of output length $\ell(n) = 1$) is conventionally defined using the notion of *nonapproximability* instead of indistinguishability.

Definition 2.2 (weak quantum hardcore) Let f be any length-regular function. A polynomial-time computable function h with length function $\ell(n)$ is called a *weak quantum hardcore (function)* of f if, for any polynomial-time quantum algorithm $\mathcal{A}$, for any polynomial p, and for any sufficiently large number $n \in \mathbb{N}$,

$$\left|\mathrm{Prob}_{x\in\{0,1\}^n,z\in\{0,1\}^{\ell(n)},\mathcal{A}}[\mathcal{A}(f(x),z,1^n)=1] - \mathrm{Prob}_{x\in\{0,1\}^n,\mathcal{A}}[\mathcal{A}(f(x),h(x),1^n)=1]\right| < \frac{1}{p(n)},$$

where x is uniformly distributed over $\{0,1\}^n$ and the subscript $\mathcal{A}$ is a random variable determined by measuring the final state of $\mathcal{A}$ in the computational basis.

Definition 2.3 (strong quantum hardcore) Let f be any length-regular function. A polynomial-time computable function h with length function $\ell(n)$ is called a *strong quantum hardcore (function)* of f if, for any polynomial-time quantum algorithm $\mathcal{A}$, for any polynomial p, and for any sufficiently large number $n \in \mathbb{N}$,

$$\left|\mathrm{Prob}_{x\in\{0,1\}^n,\mathcal{A}}[\mathcal{A}(f(x),1^n)=h(x)] - 1/2^{\ell(n)}\right| < \frac{1}{p(n)},$$

where x is uniformly distributed over $\{0,1\}^n$ and the subscript $\mathcal{A}$ is a random variable determined by measuring the final state of $\mathcal{A}$ in the computational basis.

The different between weak hardcores and strong hardcores is obvious. Although both notions coincide for hardcore functions of output length $O(\log n)$, weak hardcore functions are not always strong hardcores (see, e.g., Exercise 31 in [14]). Since any strong quantum hardcore function is also a weak quantum hardcore,

we discuss only strong hardcore functions for our purpose and drop the word "strong" in the rest of this paper for readability.

Furthermore, we are interested only in the property that a function h becomes a quantum hardcore of *any* quantum one-way function f' (of the form $f'(x,r) = (f(x),r)$). Succinctly, we refer to this property as the *quantum hardcore property* of h. Although any quantum hardcore of a "fixed" function f is also a classical hardcore of f, there is no known connection between the quantum hardcore property and the classical hardcore property.

3 How can We Prove the Quantum Hardcore Property?

We outline our argument of proving the quantum hardcore property of a given function. To prove new quantum hardcores, we exploit the notion of quantum list-decoding as a technical tool. Our approach toward list-decoding is, however, *complexity-theoretical* in nature rather than information-theoretical. Our main objects of quantum list-decoding are "classical" block codes and their codewords, which can be manipulated in a quantum fashion. Generally speaking, a block (error-correcting) *code* is a set of strings of the same length over a finite alphabet. Each string in a code is indexed by a message and is called a *codeword*. For our purpose, we are focused on a *family of codes*, which is specified by a series $\{(\Sigma_n, I_n, \Gamma_n)\}_{n\in\mathbb{N}}$ of *message space* Σ_n, *index set* I_n, and *code alphabet* Γ_n associated with a length parameter n. For convenience, we write Σ^* for the set $\bigcup_{n\in\mathbb{N}} \Sigma_n$.

As standard now in computational complexity theory, we view the code C as a function that, for each *message length* n (which serves as a *basis parameter* in this paper), maps $\Sigma_n \times I_n$ to Γ_n. We sometimes write $C^{(n)}$ to denote the code C restricted to messages of length n. Notationally, we set $N(n) = |\Sigma_n|$ and $q(n) = |\Gamma_n|$. It is convenient to assume that $\Sigma_n = (\Sigma'_n)^n$ so that n actually represents the *length* of a message over a message alphabet Σ'_n. In most cases, we simply use Γ_n as the message alphabet Σ'_n. By abbreviating $C(x,y)$ as $C_x(y)$, we also treat $C_x(\cdot)$ as a function mapping I_n to Γ_n. Denote by $M(n)$ the *block length* $|I_n|$ of each codeword. We simply set $I_n = \{0,1,\ldots,M(n)-1\}$, each element of which can be expressed in $\lceil \log_2 M(n) \rceil$ bits. We freely identify C_x with the vector $(C_x(0), C_x(1), \cdots, C_x(M(n)-1))$ in the *ambient space* $(\Gamma_n)^{M(n)}$ of dimension $M(n)$. We often work on a finite field and it is convenient to regard Γ_n as the finite field $\mathbb{F}_{q(n)}$ $(= \mathrm{GF}(q(n)))$ of order $q(n)$, provided that $q(n)$ is a prime power. The *(Hamming) distance* $d(C_x, C_y)$ between two codewords C_x and C_y is the number of non-zero components in the vector $C_x - C_y$. The *minimal distance* $d(C)$ of a code C is the smallest distance between any pair of distinct codewords in C. In contrast, $\Delta(C_x, C_y)$ denotes the *relative (Hamming) distance* $d(C_x, C_y)/M(n)$. The above-described code is simply called a $(M(n), n)_{q(n)}$-code[‡] (or $(M(n), n, d(n))_{q(n)}$-code if $d(n)$ is emphasized). We may drop a length parameter n whenever we discuss a set of codewords for a "fixed" length n; for instance, write Γ and M respectively for Γ_n and $M(n)$.

Now, let $C(x,r)$ be a function mapping $\Sigma_n \times I_n$ to Γ_n with $M(n) = p(n)$ for a certain polynomial p. We wish to prove that this function $C(x,r)$ is indeed a quantum hardcore for any quantum one-way function f' of the form $f'(x,r) = (f(x),r)$ with $|r| = p(|x|)$. For simplicity, we assume that all the elements in I_n, Σ_n, and Γ_n are expressed in binary using an appropriate, simple, easy encoding scheme. To lead to a contradiction, we first assume to the contrary that there exists a polynomial-time quantum algorithm $\mathcal{A}$ that approximates $C_x(r)$ from input $(f(x),r)$ with probability at least $1/q(n) + \varepsilon(n)$ (where $\varepsilon(n)$ is a certain *noticeable function*[§]). To be more precise, the final configuration of the quantum algorithm $\mathcal{A}$ on input (y,r), where $r \in I_n$ and $y = f(x)$ for a certain $x \in \Sigma_n$, can be assumed to be of the form:

$$\alpha_{y,r,C_x(r)}|r\rangle|C_x(r)\rangle|\phi_{y,r,C_x(r)}\rangle + \sum_{s\in\Gamma_n - \{C_x(r)\}} \alpha_{y,r,s}|r\rangle|s\rangle|\phi_{y,r,s}\rangle$$

for certain amplitudes $\alpha_{y,r,s}$ and ancilla quantum states $|\phi_{x,r,s}\rangle$ of $\ell(n)$ qubits, where the second register corresponds to the output of the algorithm, where $\ell(n)$ is a polynomially-bounded function. For each fixed y, the (restricted) algorithm $\mathcal{A}_y(\cdot) =_{def} \mathcal{A}(y,\cdot)$ gives rise to the oracle $\tilde{O}_{\mathcal{A}_y}$ (seen as a unitary operator) defined by the following transformation:

$$\tilde{O}_{\mathcal{A}_y}|r\rangle|u\rangle|t\rangle = \sum_{s\in\Gamma_n} \alpha_{y,r,s}|r\rangle|u\oplus s\rangle|t\oplus\phi_{y,r,s}\rangle$$

[‡]In some literature, the notation $(M(n), N(n))_{q(n)}$ is used instead.

[§]A function μ from $\mathbb{N}$ to $\mathbb{R}$ is said to be *noticeable* if there exists a positive polynomial p such that $\mu(n) \geq 1/p(n)$ for any sufficiently large $n \in \mathbb{N}$.

for every triplet (r,u,t) of strings, where $\oplus$ is the bitwise XOR and the notation $|t\oplus\phi_{y,r,s}\rangle$ is shorthand of the quantum state $\sum_{v:|v|=|t|}\langle v|\phi_{y,r,s}\rangle|t\oplus v\rangle$. This oracle $\tilde{O}_{\mathcal{A}_y}$ describes *computational error* (not transmission error) occurring during the computation of C_x by the (possibly) faulty quantum algorithm $\mathcal{A}$. This type of erroneous quantum computation is similar to the computational errors (e.g., [1, 3, 4, 23]) dealt with in quantum computational cryptography and quantum algorithm designing. Remember that all the amplitude $\{\alpha_{y,r,s}\}_{r,s,}$ in $\tilde{O}_{\mathcal{A}_y}$ could be chosen adversely, not favorably, to us. Since $\tilde{O}_{\mathcal{A}_y}$ is a unitary operation, its inverse $\tilde{O}_{\mathcal{A}_y}^{-1}$ can be uniquely defined. Given the oracle $\tilde{O}_{\mathcal{A}_y}$, we can freely access $\tilde{O}_{\mathcal{A}_y}$ as well as $\tilde{O}_{\mathcal{A}_y}^{-1}$ by simply invoking a query, using three registers containing (r,u,t). Upon an oracle call, the oracle is automatically applied to the three registers and all the contents of these registers are modified at the cost of unit time.

Similar to the classical notion of a *received word* in coding theory, we introduce our terminology concerning an oracle that represents a "quantum-computationally" corrupted codeword that produces garbage information of $\ell(n)$ size. For an immediate comparison to a quantum case, we use a more conceptual term "classically corrupted codeword" instead of the conventional term "received word" in the rest of this paper.

Definition 3.1 (quantum-computationally corrupted codeword) We say that an oracle $\tilde{O}$ *represents a quantum-computationally (or quantumly) corrupted codeword* if there exists a function ℓ mapping $\mathbb{N}$ to $\mathbb{N}$ such that, for any length parameter $n\in\mathbb{N}$, any index $r\in I_n$, any symbol $u\in\Gamma_n$, and any modifier $t\in\{0,1\}^{\ell(n)}$, the oracle $\tilde{O}$ satisfies $\tilde{O}|r\rangle|u\rangle|t\rangle=\sum_{s\in\Gamma_n}\alpha_{r,s}|r\rangle|u\oplus s\rangle|t\oplus\phi_{r,s}\rangle$ for certain complex numbers $\alpha_{r,s}$ and unit vectors $|\phi_{r,s}\rangle$ in a $2^{\ell(n)}$-dimensional Hilbert space, depending only on (r,s). The parameter $\ell(n)$ indicates the size of *garbage information*. Clearly, $\tilde{O}$ is a unitary operator acting on a Hilbert space spanned by the elements of $\bigcup_{n\in\mathbb{N}}(I_n\times\Gamma_u\times\{0,1\}^{\ell})$. For convenience, we identify a quantumly corrupted codeword with its representing oracle and we simply call $\tilde{O}$ a quantumly corrupted codeword.

To lead to our desired contradiction, we need to invert the function $f(x)$ by extracting x from the aforementioned quantumly corrupted codeword $\tilde{O}_{\mathcal{A}}$ in time polynomial in $|x|$. Fix $n\in\mathbb{N}$ and $x\in\Sigma_n$. Consider the entity $(1/M(n))\sum_{r\in I_n}|\alpha_{r,C_x(r)}|^2$ that yields the probability of $\mathcal{A}$'s computing $C_x(\cdot)$ correctly on average. This entity also indicates "closeness" between the codeword C_x and the quantumly corrupted codeword $\tilde{O}_{\mathcal{A}}$. In classical list-decoding, for any given oracle $\tilde{O}$ that represents a *classically corrupted codeword* and for any error bound ε, we need to output a list that include all messages x for which the probability over $r\in I_n$ that $\tilde{O}(r)$ equals $C_x(r)$ is at most $1-\varepsilon$ (namely, $\mathrm{Prob}_{r\in I_n}[\tilde{O}(r)=C_x(r)]\geq 1-\varepsilon$). By setting $p_{r,s}=1$ if $\tilde{O}(r)=s$ and 0 otherwise, the behavior of $\tilde{O}$ can be viewed in a style of unitary operation as $\tilde{O}|r\rangle|0\rangle=\sum_{r\in I_n}p_{r,s}|r\rangle|s\rangle$. The aforementioned entity $(1/M(n))\sum_{r\in I_n}|\alpha_{r,C_x(r)}|^2$ equals the probability $\mathrm{Prob}_{r\in I_n}[\tilde{O}(r)=C_x(r)]$ in a classical setting. For our convenience, we name this entity the *presence* of C_x in $\tilde{O}$ and denote it by $\mathrm{Pre}_{\tilde{O}}(C_x)$. The requirement for the error rate of classical list-decoding is therefore rephrased as $\mathrm{Pre}_{\tilde{O}}(C_x)\geq 1-\varepsilon$.

From a slightly different view point, we argue that presence is indeed an extension of relative (Hamming) distance. This will be used in the proof of Lemma 3.2. Let v denote a classically corrupted codeword. We can view v as a binary vector in the $q(n)M(n)$-dimensional space, in which the rth block $v[r]$ of v is of the form $0^{i-1}10^{q(n)-i-2}$ for a certain index $i\in[q(n)]$, where $r\in M(n)$. Using this new representation, the relative (Hamming) distance between two classically corrupted codewords v and w equals the ℓ_1-norm $\|v-w\|_1=\sum_{r\in[M(n)]}\|v[r]-w[r]\|$. Similarly, for a quantumly corrupted codeword $v_{\tilde{O}}$ that an oracle $\tilde{O}$ represents, $v_{\tilde{O}}$ can be viewed as the real vector in the $q(n)M(n)$-dimensional space, in which $v_{\tilde{O}}[r]$ is $(|\alpha_{r,1}|^2,|\alpha_{r,2}|^2,\ldots,|\alpha_{r,q(n)}|^2)$. The presence $\mathrm{Pre}_{\tilde{O}}(C_x)$ now indicates the ℓ_1-norm between $v_{\tilde{O}}$ and a codeword C_x, extending the classical notion of distance.

With the above notions, we formulate a quantum version of a classical list-decoding problem. Recall that our function $C(x,r)$ can be treated as an $(M(n),n,d(n))_{q(n)}$-code family $C=\{C_x\}_{x\in\Sigma^*}$. Let $\varepsilon(n)$ be any *error bias parameter*.

ε-Quantum List Decoding Problem (ε-QLDP) for Code C

Input: a message length n, and a value $1/\varepsilon(n)$ that is expressed in binary.
Implicit Input: an oracle $\tilde{O}$ representing a quantumly corrupted codeword of arbitrary garbage size.

Output: a list of messages including all messages $x\in\Sigma_n$ such that $\mathrm{Pre}_{\tilde{O}}(C_x)\geq 1/q(n)+\varepsilon(n)$; in other words, codewords C_x have "slightly" higher presence in $\tilde{O}$ than the average. For convenience, we call such a list a *valid list* for the ε-QLDP.

Our formulation of the problem ε-QLDP deals with any quantumly corrupted codewords of arbitrary garbage size ℓ. This formalism solely stems from our target of quantum hardcores. Therefore, for other

applications, we may possibly bound the garbage size of quantumly corrupted codewords by, e.g., certain "fixed" functions.

Because no polynomial-time quantum list-decoder D can output a valid list of super-polynomial size for any given quantumly corrupted codeword $\tilde{O}$, there is an important question to answer: how many messages x satisfy the required inequality $\mathrm{Pre}_{\tilde{O}}(C_x) \geq 1/q(n)+\varepsilon(n)$? We want to show an upper bound of the number of codewords that have relatively high presence in a given quantumly corrupted word. For our proof, we employ a proof method of Guruswami and Sudan [19], who gave a q-ary extension of *Johnson bound* using a geometric method.

Lemma 3.2 *Let n be any message length. Let $\varepsilon(n)$, $q(n)$, $d(n)$, and $M(n)$ satisfy that $\varepsilon(n) > \ell(n) =_{def} (1-1/q(n))\sqrt{1-d(n)/M(n)\,(1+1/(q(n)-1))}$. For any $(M(n),n,d(n))_{q(n)}$-code C and for any quantumly corrupted codeword $\tilde{O}$, there are at most*

$$J_{\varepsilon,q,d,M}(n) =_{def} \min\left\{M(n)(q(n)-1), \frac{d(n)\,(1-1/q(n))}{d(n)\,(1-1/q(n)) + M(n)\varepsilon^2(n) - M(n)\,(1-1/q(n))^2}\right\}$$

messages $x \in \Sigma_n$ such that $\mathrm{Pre}_{\tilde{O}}(C_x) \geq 1/q(n)+\varepsilon(n)$. If $\varepsilon(n) = \ell(n)$, then the above bound can be replaced by $2M(n)(q(n)-1)-1$.

The proof of Lemma 3.2 is in essence an elaborated modification of the proof in [19]. For completeness, we include the detailed proof of the lemma in Appendix. Here, as an example, we give the value $J_{\varepsilon,q,d,M}(n)$ for a $(q^n, n, q^n - q^{n-1})_q$ Hadamard code.

Example: Hadamard Codes. Consider an $(M(n),n,d(n))_{q(n)}$ Hadamard code $\mathrm{HAD}^{(q)} = \{\mathrm{HAD}^{(q)}_x\}_{x\in\Sigma^*}$ with $M(n) = q(n)^n$ and $d(n) = (1-1/q(n))\,M(n)$. Assume that our bias parameter ε is non zero (i.e., $\epsilon(n) > 0$ for all $n \in \mathbb{N}$). Lemma 3.2 guarantees that, for any quantumly corrupted codeword $\tilde{O}$, the number of codeword candidates that satisfy the inequality $\mathrm{Pre}_{\tilde{O}}(\mathrm{HAD}^{(q)}_x) \geq 1/q(n)+\varepsilon(n)$ is at most

$$\frac{d(n)\left(1-\frac{1}{q(n)}\right)}{d(n)\left(1-\frac{1}{q(n)}\right) + M(n)\epsilon(n)^2 - M(n)\left(1-\frac{1}{q(n)}\right)^2} = \left(1-\frac{1}{q(n)}\right)^2 \cdot \frac{1}{\epsilon(n)^2}.$$

In particular, if there exists a positive polynomial p satisfying $\epsilon(n) \geq 1/p(n)$ for all $n \in \mathbb{N}$, there are only at most $(1-1/q(n))^2\,p(n)$ codeword candidates.

Let us return to our argument. We use the term "quantum list decoding algorithms" (or "quantum list decoders") to mean a procedure of solving the problem ε-QLDP with a specific *confidence parameter* $\delta(n)$. Formally, we define this notion as follows.

Definition 3.3 (quantum list decoder) Let C be any code, let $\varepsilon(n)$ be any error bias, and let $\delta(n)$ be any confidence parameter. Any quantum algorithm (i.e., a unitary operator) $\mathcal{D}$ that solves the ε-QLDP for C with success probability at least $\delta(n)$ is called a *quantum list-decoding algorithm* for C with respect to (ε,δ) (or (ε,δ)-quantum list-decoder). If $\mathcal{D}$ further runs in time polynomial in $(n, 1/\varepsilon(n), 1/(1-\delta(n)))$ (if $\delta(n) = 1$ then we treat $1/(1-\delta(n))$ as 1 for notational convenience), it is called a *polynomial-time quantum list-decoding algorithm* for C with respect to (ε,δ). We also say that C is (ε,δ)-*quantum list-decodable* if C has an (ε,δ)-quantum list-decoder.

For simplicity, we assume that, for each oracle access, our quantum list-decoder $\mathcal{D}$ uses its last three registers $|r\rangle|u\rangle|t\rangle$, in which the last register holds an arbitrarily large quantum state.

For convenience, we assume that, after an oracle call, the oracle $\tilde{O}$ (or its inverse $\tilde{O}^{-1}$) is automatically applied to the last three registers of D with a unit cost of time although the last register for garbage information that can be produced by $\tilde{O}$ may be extremely long. For convenience, the last register is assumed to hold only 0s at the beginning of the computation.

Now, we wish to complete our argument (which we started at the beginning of this section). Let us assume that there exists a polynomial-time quantum list-decoding algorithm D that solves the ε-QLDP for $C_x(\cdot)$ with certain noticeable probability, say $\delta(n)$. Meanwhile, we assume that the aforementioned quantumly corrupted codeword $\tilde{O}_{\mathcal{A}_y}$ of garbage size $\ell(n)$ can be realized by appropriately executing $\mathcal{A}$. With oracle access to this oracle $\tilde{O}_{\mathcal{A}_y}$ (as well as its inverse $\tilde{O}^{-1}_{\mathcal{A}_y}$), this quantum list-decoder D can produce with probability at least $\delta(n)$ all possible candidates $x' \in f^{-1}(y)$ that have the required presence $\mathrm{Pre}_{\tilde{O}_{\mathcal{A}_y}}(C_{x'})$ at least $1/q(n)+\varepsilon(n)$. Since we can check whether x' belongs to the inverse image $f^{-1}(y)$ in polynomial

time, our quantum list-decoder D gives rise to a polynomial-time quantum algorithm that inverts f with noticeable probability on average. This clearly contradicts the quantum one-wayness of f.

Can we realize $\tilde{O}_{\mathcal{A}_y}$ using $\mathcal{A}$? Unfortunately, it is not clear that there is a generic unitary procedure of converting $\mathcal{A}$ to $\tilde{O}_{\mathcal{A}_y}$. For our proof, we instead use an alternative oracle $\tilde{O}$, which is a slight modification of $\tilde{O}_{\mathcal{A}_y}$. Let us assume that the quantum list-decoder D runs in time $p(n)$ for a certain polynomial p. From this time bound, we understand that D cannot access more than the first $p(n)$ qubits of the content of the last register. Recall that $|\phi_{y,r,s}\rangle$ is a quantum state of $\ell(n)$-qubits. Let $m(n) = \ell(n) + \lceil \log_2 q(n) \rceil$ for a certain polynomial ℓ. We want to define another quantumly corrupted codeword $\tilde{O}$ of garbage size $p(n)^2$. The following algorithm describes the behavior of $\tilde{O}$. Let $j \in [\ell, p(n)^2]_{\mathbb{Z}}$ be an arbitrary integer and let $t \in \Sigma^j$.

> Given input $|r\rangle|u\rangle|t, 0^{p(n)^2-j}\rangle$, we swap the registers and compute $|u\rangle|t\rangle|0^{p(n)^2-j}\rangle \otimes \mathcal{A}_y|r\rangle|0^{m(n)}\rangle$. We then obtain $|u\rangle|t\rangle|0^{p(n)^2-j}\rangle \otimes \sum_s \alpha_{y,r,s}|r\rangle|s\rangle|\phi_{y,r,s}\rangle$. Assume that $|\phi_{y,r,s}\rangle$ is of the form $\sum_{w\in\{0,1\}^{\ell(n)}} \beta_w^{(y,r,s)}|w\rangle$. Transform $|u\rangle|t\rangle|s\rangle|w, 0^{m(n)}\rangle$ to $|u \oplus s\rangle|t \oplus w0^{j-\ell(n)}\rangle|s, w\rangle$. Rearrange the registers and we output the quantum state
>
> $$\sum_{s\in\Gamma_n} \alpha_{y,r,s}|r\rangle|u \oplus s\rangle|(t0^{m(n)}) \oplus \hat{\phi}^{(j)}_{y,r,s}\rangle|0^{p(n)^2-j-m(n)}\rangle,$$
>
> where $|\hat{\phi}^{(j)}_{y,r,s}\rangle = \sum_{w\in\{0,1\}^{\ell(n)}} |w0^{j-\ell(n)}\rangle|s, w\rangle$.

The reader should check that this new oracle $\tilde{O}$ in essence realizes $\tilde{O}_{\mathcal{A}_y}$. We first claim that, after the ith oracle call (to either $\tilde{O}$ or $\tilde{O}^{-1}$), the last register of D contains a quantum state of the form $\sum_{t\in\{0,1\}^{ip(n)}} \gamma_t|t\rangle|0^{p(n)^2-ip(n)}\rangle$. This can be shown as follows. We consider the case of an oracle call to $\tilde{O}$. At the time of the ith oracle call, D makes a query of the form $|r\rangle|u\rangle|t0^{p(n)^2-ip(n)}\rangle$ with $t \in \{0,1\}^{ip(n)}$. In response to the oracle call, the oracle $\tilde{O}$ produces a quantum state whose last register contains terms of the form $|(t0^{p(n)^2-ip(n)}) \oplus \hat{\phi}^{(ip(n))}_{y,r,s}\rangle|0^{p(n)^2-ip(n)-m(n)}\rangle$. Clearly, $p(n)^2 - ip(n) - m(n) \leq p(n)^2 - (i+1)p(n)$. In the case of the inverse $\tilde{O}^{-1}$, the definition of $\tilde{O}$ guarantees that the query $|r\rangle|u\rangle|t0^{p(n)^2-ip(n)}\rangle$ can be transformed into a quantum state whose last register is of the form $|\psi\rangle|0^{p(n)^2-ip(n)}\rangle$.

Because D should work even with this new oracle $\tilde{O}$, producing all the possible candidates x' in polynomial time. Therefore, we obtain the following key theorem that bridges between quantum hardcores and quantum list-decoding. This theorem serves as a driving force to develop a theory of quantum list decoding in the subsequent sections.

Theorem 3.4 *Let $C = \{C_x\}_{x\in\Sigma^*}$ be any $(M(n), n, d(n))_{q(n)}$-code with a message space $\Sigma^* = \bigcup_{n\in\mathbb{N}} \Sigma_n$, which is polynomial-time computable, where $\log_2 M(n) \in n^{O(1)}$ and $\log_2 q(n) \in n^{O(1)}$. If, for any noticeable function $\varepsilon(n)$, there exist a noticeable function $\delta(n)$ and a polynomial-time (ε, δ)-quantum list-decoder for C, then $C(x, r)$ is a quantum hardcore for any quantum one-way function f' of the form $f'(x, r) = (f(x), r)$ with $|x| = \lceil \log_2 N(n) \rceil$ and $|r| = \lceil \log_2 M(n) \rceil$, where $N(n) = |\Sigma_n|$.*

The rest of this paper is devoted to the construction of quantum list-decoders for each given quantum hardcore candidate. The first step is to establish a generic technique of constructing quantum list-decoders for "well-behaved" classical block codes.

4 How can We Construct Quantum List-Decoding Algorithms?

Theorem 3.4 gives a sufficient condition to prove the quantum hardcore property of any given function h. It is therefore enough for us to design a quantum algorithm that solves the QLDP for h with high probability in polynomial time. Our task in this paper is to find a generic way to construct a polynomial-time quantum list-decoder for a wide range of classical block codes. Classically, however, it seems hard to design such list-decoding algorithms in general. Nevertheless, a robust nature of quantum computation enables us to prove that, as long as we have a decoding algorithm $\mathcal{A}$ from a unique quantum state (called a *codeword state*), we can construct a quantum list-decoder by calling $\mathcal{A}$ as a black-box oracle. Definition 4.1 formally introduces such quantum states. In Section 7, the notion of such codeword states plays a central role as our technical tool in proving new quantum hardcores.

Hereafter, we assume the basic arithmetic operations (multiplication, addition, subtraction, division, etc.) on a finite field $\mathbb{F}_q$ of order q. When q is a prime number, $\mathbb{F}_q$ can be identified with the integer

ring $\mathbb{Z}/q\mathbb{Z}$ whose elements are written as $0, 1, 2, \ldots, q-1$. For convenience, let $\mathbb{F}_q^+$ stand for $\mathbb{F}_q - \{0\}$. The notation $\mathbb{Z}_q$ denotes the (finite) permutation group whose elements are $0, 1, 2, \ldots, q-1$. Moreover, we write $[m, n]_{\mathbb{Z}} = \{m, m+1, m+2, \ldots, n\}$ for any two integers $m, n \in \mathbb{N}$ with $m \leq n$ and, in particular, let $[q] = [1, q]_{\mathbb{Z}}$ for any integer $q \geq 1$ in the rest of this paper. Finally, we denote by ω_q the complex number $e^{2\pi\iota/q}$, where e is the base of natural logarithms and $\iota = \sqrt{-1}$.

Definition 4.1 (shuffled codeword state) Let $C = \{C_x\}_{x\in\Sigma^*}$ be any $(M(n), n)_{q(n)}$-code family with a message space $\Sigma^* = \bigcup_{n\in\mathbb{N}} \Sigma_n$ and a series $\{I_n\}_{n\in\mathbb{N}}$ of index sets. Let k be any element in $\mathbb{F}_{q(n)}^+$. A *k-shuffled codeword state* for the codeword C_x that encodes a message $x \in \Sigma_n$ is the quantum state

$$|C_x^{(k)}\rangle = \frac{1}{\sqrt{M(n)}} \sum_{r\in I_n} \omega_{q(n)}^{k\cdot C_x(r)} |r\rangle.$$

In particular, when $k = 1$, we use the simplified notation $|C_x\rangle$ for $|C_x^{(1)}\rangle$.

The reader may be aware that our notion of codeword states is not anew; the codeword states for certain binary codes have already appeared implicitly in several important quantum algorithms. For instance, Grover's search algorithm [18] produces such a codeword state after the first oracle call. In the quantum algorithms of Bernstein and Vazirani [6], of Deutch and Jozsa [12], and of van Dam, Hallgren, and Ip [28], such codeword states are generated to obtain their results. All these quantum algorithms hinge at generating codeword states.

Next, we discuss how to generate the k-shuffled codeword state $|C_x^{(k)}\rangle$ for each q-ary codeword C_x with oracle accesses to a quantumly corrupted codeword $\tilde{O}$. It is rather straightforward to generate the quantum state $|C_x\rangle$ from the oracle O_{C_x} that represents C_x *without* any corruption (behaving as the "standard" oracle). Here, we claim that there exists a generic quantum algorithm that generates codeword states for any q-ary code C.

Theorem 4.2 *Let C be any $(M(n), n)_{q(n)}$-code family with a message space $\Sigma^* = \bigcup_{n\in\mathbb{N}} \Sigma_n$, where $q(n)$ is a prime number for every $n \in \mathbb{N}$. Let m be any function from $\mathbb{N}$ to $\mathbb{N}$. There exists a quantum algorithm $\mathcal{A}$ that, for any message length $n \in \mathbb{N}$, for any quantumly corrupted codeword $\tilde{O}$ with garbage size $\ell(n)$, for any message $x \in \Sigma_n$, and for any $k \in \mathbb{F}_{q(n)}^+$, generates the quantum state*

$$|\psi_k\rangle = \kappa_x^{(k)} |k\rangle |C_x^{(k)}\rangle |\tau\rangle + |\Lambda_x^{(k)}\rangle$$

from the initial state $|\psi_k^{(0)}\rangle = |k\rangle|0^{\lceil \log_2 M(n)\rceil}\rangle|0\rangle|0^{\ell(n)}\rangle$ with only two queries to $\tilde{O}$ and $\tilde{O}^{-1}$, where $|\tau\rangle$ is a fixed basis vector, and $\kappa_x^{(k)}$ is a complex number, and $|\Lambda_x^{(k)}\rangle$ is a vector satisfying $(\langle k|\langle C_x^{(k)}|\langle\tau|)|\Lambda_x^{(k)}\rangle = 0$ with the following condition: for every $x \in \Sigma_n$, there exists an element $k \in \mathbb{F}_{q(n)}^+$ with the inequality $|\kappa_x^{(k)}| \geq (q(n)/(q(n)-1))\,|\mathrm{Pre}_{\tilde{O}}(C_x) - 1/q(n)|$. Moreover, $\mathcal{A}$ runs in time polynomial in $(n, \log_2 q(n), \log_2 M(n))$.

When $q = 2$, the bound of $|\kappa_x^{(1)}|$ in Theorem 4.2 matches the bound of Adcock and Cleve [1]. In the theorem, if the code family $C = \{|C_x\rangle\}_{x\in I_n}$ further satisfies "orthogonality," which we call *phase orthogonality* of C, we can isolate simultaneously all individual messages x, as in Corollary 4.5. Phase-orthogonality for a binary code, in particular, is naturally induced from the standard *inner product* of two codewords when we translate their binary symbols $\{0, 1\}$ into $\{+1, -1\}$.

Definition 4.3 (phase orthogonality) A classic block code $C = \{C_x\}_{x\in\Sigma^*}$ is called *k-shuffled phase orthogonal* if $\langle C_x^{(k)}|C_y^{(k)}\rangle = 0$ for any $n \in \mathbb{N}$ and any two distinct messages $x, y \in \Sigma_n$. If C is k-shuffled phase-orthogonal for every element $k \in \mathbb{F}_q^+$, then C is simply called *phase orthogonal*.

Since k-shuffled codeword states are pure quantum states, the value $|\langle C_x^{(k)}|C_y^{(k)}\rangle|$ coincides with the *fidelity* $F(|C_x^{(k)}\rangle, |C_y^{(k)}\rangle)$ of $|C_x^{(k)}\rangle$ and $|C_y^{(k)}\rangle$. The next lemma relates this fidelity $F(|C_x^{(k)}\rangle, |C_y^{(k)}\rangle)$ to the relative Hamming distance $\Delta(C_x, C_y)$. It is not difficult to prove the following lemma, which will be used in Section 7.

Lemma 4.4 *For any pair (C_x, C_y) of codewords in a given $(M(n), n, d(n))_{q(n)}$-code C and for any $k \in \mathbb{F}_q^+$,*

$$F(|C_x^{(k)}\rangle, |C_y^{(k)}\rangle) \geq 1 - 2\Delta(C_x, C_y),$$

where the equality holds for any binary code C.

Proof. Fix n arbitrarily and drop the subscript "n" for simplicity. For each index $\ell \in [0, q-1]_{\mathbb{Z}}$, we define $d_\ell^{(k)}(C_x, C_y) = |\{r \in \mathbb{F}_M \mid k(C_x(r) - C_y(r)) = \ell \bmod q\}|$. Since $F(|C_x^{(k)}\rangle, |C_y^{(k)}\rangle) = |\langle C_x^{(k)}|C_y^{(k)}\rangle|$, it follows that

$$\begin{aligned} F(|C_x^{(k)}\rangle, |C_y^{(k)}\rangle) &= \left|1 - \frac{1}{M} \cdot \omega_q^0 \cdot d_0^{(k)}(C_x, C_y) + \frac{1}{M} \sum_{\ell=1}^{q-1} \omega_q^\ell \cdot d_\ell^{(k)}(C_x, C_y)\right| \\ &\geq \left|1 - \frac{d(C_x, C_y)}{M}\right| - \frac{1}{M} \sum_{\ell=1}^{q-1} \left|\omega_q^\ell \cdot d_\ell^{(k)}(C_x, C_y)\right| \\ &\geq (1 - \Delta(C_x, C_y)) - \Delta(C_x, C_y) \;=\; 1 - 2\Delta(C_x, C_y), \end{aligned}$$

which gives the desired bound of the lemma. In particular, when $q = 2$, since $d_1^{(k)}(C_x, C_y) = d(C_x, C_y)$ and $\omega_2 = -1$, we obtain the equality $F(|C_x^{(k)}\rangle, |C_y^{(k)}\rangle) = 1 - 2\Delta(C_x, C_y)$. □

One of the benefits of phase-orthogonality is explained in the following corollary. If a code $C = \{C_x\}_{x \in \Sigma^*}$ is phase orthogonal, then the set $\{|C_x^{(k)}\rangle\}_{x \in \Sigma_n}$ of k-shuffled codeword states, for each $n \in \mathbb{N}$ and each $k \in \mathbb{F}_{q(n)}^+$, forms an orthnormal basis of an $M(n)$-dimensional Hilbert space. Hence, we obtain the following corollary of Theorem 4.2.

Corollary 4.5 *Let $n \in \mathbb{N}$ and let $C = \{C_x\}_{x \in \Sigma_n}$ be any phase-orthogonal $(M(n), n)_{q(n)}$-code. There exists a quantum algorithm $\mathcal{A}$ that, for each message length $n \in \mathbb{N}$, starting with $|\phi^{(0)}\rangle = |0\rangle|0^{\lceil \log_2 M(n) \rceil}\rangle|0\rangle|0^{\ell(n)}\rangle$ with any quantumly corrupted codeword $\tilde{O}$ with respect to a parameter $m(n)$, $\mathcal{A}$ makes only two queries to $\tilde{O}$ and $\tilde{O}^{-1}$ and generates the quantum state*

$$|\psi'\rangle = \frac{1}{\sqrt{q-1}} \sum_{k \in \mathbb{F}_{q(n)}^+} \sum_{x \in \Sigma_n} \kappa_x^{(k)} |k\rangle |C_x^{(k)}\rangle |\tau\rangle + |\Lambda'\rangle$$

such that, for every message $x \in \Sigma_n$, there exists an element $k \in \mathbb{F}_{q(n)}^+$ satisfying that $|\kappa_x^{(k)}| \geq (q(n)/(q(n)-1))\,|\mathrm{Pre}_{\tilde{O}}(C_x) - 1/q(n)|$ and $(\langle k|\langle C_x^{(k)}|\langle \tau|)|\Lambda'\rangle = 0$.

Now, we give the proof of our key theorem, Theorem 4.2. Notice that the theorem is true for any q-ary code. The binary case ($q = 2$) was discussed implicitly in [1]; however, our argument for the general q-ary case is more involved because of our "k-shuffledness" condition. For our proof, we assume the following limited form of quantum Fourier transform F_n, over a finite permutation group $\mathbb{Z}_n$, running in time polynomial in n: for any $s \in \mathbb{Z}_n$,

$$F_n : |s\rangle \to \frac{1}{\sqrt{n}} \sum_{r \in \mathbb{Z}_n} \omega_n^{s \cdot r} |r\rangle.$$

For a more general form of quantum Fourier transform over a finite field $\mathbb{F}_n$, see, e.g., [28].

Proof of Theorem 4.2. Since $q(n)$ is a prime number, we use $\{0, 1, 2, \ldots, q(n)-1\}$ as the elements of $\mathbb{F}_{q(n)}$. We assume the premise of the theorem. Let C be any $(M(n), n)_{q(n)}$-code family with message space $\Sigma^* = \bigcup_{n \in \mathbb{N}} \Sigma_n$, index sets $\{I_n\}_{n \in \mathbb{N}}$, and code alphabets $\{\Gamma_n\}_{n \in \mathbb{N}}$. Note that $M(n) = |I_n|$. Let $\tilde{O}$ be any quantumly corrupted codeword of garbage size $\ell(n)$ for C, where ℓ is an arbitrary function. First, we describe our quantum codeword-state generation algorithm $\mathcal{A}$ in detail. Fix $n \in \mathbb{N}$, $x \in \Sigma_n$, and $k \in \mathbb{F}_{q(n)}^+$ in the following description. For simplicity, we drop the script "n" and also let $m = \lceil \log_2 M \rceil$.

Quantum Algorithm $\mathcal{A}$:

(1) Start with the initial state $|\psi_k^{(0)}\rangle = |k\rangle|0\rangle|0\rangle|0^\ell\rangle$.
(2) Apply the quantum Fourier transform F_M to the second register, and we obtain the superposition

$$|\psi_k^{(1)}\rangle = \frac{1}{\sqrt{M}} \sum_{r \in I_n} |k\rangle|r\rangle|0\rangle|0^\ell\rangle.$$

(3) Invoke a query to the oracle $\tilde{O}$ using the last three registers. The resulting quantum state is

$$|\psi_k^{(2)}\rangle = \frac{1}{\sqrt{M}} \sum_{r \in I_n} \sum_{z \in \mathbb{F}_q} \alpha_{r,z} |k\rangle|r\rangle|z\rangle|\phi_{r,z}\rangle.$$

(4) Encode the information on the first and the third resisters into the "phase" so that we obtain the quantum state of the form

$$|\psi_k^{(3)}\rangle = \frac{1}{\sqrt{M}} \sum_{r\in I_n} \sum_{z\in\mathbb{F}_q} \omega_q^{k\cdot z} \alpha_{r,z} |k\rangle|r\rangle|z\rangle|\phi_{r,z}\rangle.$$

For convenience, this step will be referred to as *phase encoding*.

(5) Apply the inverse oracle $\tilde{O}^{-1}$ to the last three registers and denote the resulting state $(I\otimes\tilde{O}^{-1})|\psi_k^{(3)}\rangle$ by $|\psi_k^{(4)}\rangle$. See, e.g., [1] for how to implement $\tilde{O}^{-1}$ from $\tilde{O}$. Although ℓ is extremely large, when the oracle is called, the last three registers are automatically changed. There is no need of scanning all the qubits stored in these registers before halting the computation. The final state $|\psi_k^{(4)}\rangle$ can be expressed in the form $\kappa_x^{(k)}|k\rangle|C_x^{(k)}\rangle|\tau\rangle + |\Lambda_x^{(k)}\rangle$, where $|\tau\rangle = |0\rangle|0^\ell\rangle$ and $(\langle k|\langle C_x^{(k)}|\langle\tau|)|\Lambda_x^{(k)}\rangle = 0$.

End of the Algorithm

The execution time of $\mathcal{A}$ is clearly upper-bounded by a certain polynomial in $(n, \log_2 q, \log_2 M)$. Now, we want to calculate the amplitude $\kappa_x^{(k)}$. First, we note that the quantum state $|\psi'\rangle = (I\otimes\tilde{O})|k\rangle|C_x^{(k)}\rangle|\tau\rangle$ is of the form

$$|\psi'_{k,x}\rangle = \frac{1}{\sqrt{M}} \sum_{r\in I_n} \sum_{z\in\mathbb{F}_q} \omega_q^{k\cdot C_x(r)} \alpha_{r,z}^{(k)} |r\rangle|z\rangle|\phi_{r,z}\rangle.$$

Therefore, we have

$$\begin{aligned}\kappa_x^{(k)} &= (\langle k|\langle C_x^{(k)}|\langle\tau|)((I\otimes\tilde{O}^{-1})|\psi_k^{(3)}\rangle) = \langle\psi'_{k,x}|\psi_k^{(3)}\rangle \\ &= \frac{1}{M}\sum_{r\in I_n}\sum_{z\in\mathbb{F}_q} \omega_q^{k(z-C_x(r))}|\alpha_{r,z}|^2.\end{aligned}$$

The non-trivial part of our proof is to show a lower-bound of $|\kappa_x^{(k)}|$. Notice that a different proof appeared in [25]. By summing $\kappa_x^{(k)}$ over all $k\in\mathbb{F}_q^+$, the term $\sum_{k\in\mathbb{F}_q^+}|\kappa_x^{(k)}|$ is lower-bounded by

$$\sum_{k\in\mathbb{F}_q^+}\left|\kappa_x^{(k)}\right| \geq \left|\sum_{k\in\mathbb{F}_q^+}\frac{1}{M}\sum_{r\in I_n}\sum_{z\in\mathbb{F}_q}\omega_q^{k(z-C_x(r))}|\alpha_{r,z}|^2\right| = \left|\sum_{k\in\mathbb{F}_q^+}\sum_{j\in\mathbb{F}_q}\omega_q^{k\cdot j}\left(\frac{1}{M}\sum_{r\in I_n}\left|\alpha_{r,C_x(r)+j}\right|\right)\right|.$$

We introduce a notation. For each value $j\in\mathbb{F}_q$, write β_j for the term $(1/M)\sum_{r\in I_n}|\alpha_{r,C_x(r)+j}|^2$. Note that $\beta_0 = \mathrm{Pre}_{\tilde{O}}(C_x)$ and $1-\beta_0 = \sum_{j\in\mathbb{F}_q^+}\beta_j$. Using this β_j-notation, we have

$$\begin{aligned}\left|\sum_{k\in\mathbb{F}_q^+}\sum_{j\in\mathbb{F}_q}\omega_q^{k\cdot j}\beta_j\right| &= \left|\sum_{k\in\mathbb{F}_q^+}\omega_q^0\beta_0 + \sum_{k\in\mathbb{F}_q^+}\omega_q^k\beta_1 + \cdots + \sum_{k\in\mathbb{F}_q^+}\omega_q^{(q-1)k}\beta_{q-1}\right| \\ &= \left|(q-1)\beta_0 - \sum_{j\in\mathbb{F}_q^+}\beta_j\right| \\ &= |(q-1)\mathrm{Pre}_{\tilde{O}}(C_x) - (1-\mathrm{Pre}_{\tilde{O}}(C_x))| \\ &= |q\cdot\mathrm{Pre}_{\tilde{O}}(C_x) - 1|.\end{aligned}$$

Hence, we obtain $(1/(q-1))\sum_{k\in\mathbb{F}_q^+}|\kappa_x^{(k)}| \geq (1/(q-1))|q\cdot\mathrm{Pre}_{\tilde{O}}(C_x)-1|$. This implies that there exists a number $k\in\mathbb{F}_q^+$ for which

$$\left|\kappa_x^{(k)}\right| \geq \frac{1}{q-1}|q\cdot\mathrm{Pre}_{\tilde{O}}(C_x)-1| = \frac{q}{q-1}\left|\mathrm{Pre}_{\tilde{O}}(C_x) - \frac{1}{q}\right|.$$

This completes the proof. □

Theorem 4.2 provides a generic way of generating a k-shuffled codeword state $|C_x^{(k)}\rangle$ from $\tilde{O}$. Our next task is to recover x with reasonable probability from each k-shuffled codeword state $|C_x^{(k)}\rangle$. Any quantum algorithm that completes this task is succinctly called a *codeword-state decoder*. We formally define the quantum codeword-state decoders.

Definition 4.6 (quantum codeword-state decodability) Let $\eta \in [0,1]$. A classical $(M(n),n)_{q(n)}$-code family is said to be *η-quantum codeword-state decodable* if there exists a quantum algorithm that, on input $n \in \mathbb{N}$ and $k \in \mathbb{F}_{q(n)}^{+}$ as well as $|C_x^{(k)}\rangle$, recovers x with success probability at least η. Such an algorithm is simply called an *η-quantum codeword-state decoder*.

The following theorem, which is general but slightly technical, shows how to convert a quantum codeword state into a quantum list-decoder. This complements Theorem 4.2. Recall the definition of $J_{\varepsilon,q,d,M}(n)$ given in Lemma 3.2. Note that $J_{\varepsilon,q,d,M}(n) \leq 2M(n)q(n)$.

Theorem 4.7 *Let $C = \{C_x\}_{x\in\Sigma^*}$ be any $(M(n),n,d(n))_{q(n)}$-code. Let $\mathcal{A}$ denote the quantum algorithm given in Theorem 4.2. Let ε,δ be any two nonnegative functions $\varepsilon(n)$ and $\delta(n)$ with $0 \leq \varepsilon(n) \leq 1$ and $0 \leq \delta(n) < 1$ for all $n \in \mathbb{N}$. If there exists a $(1-\nu(n))$-quantum codeword-state decoder $\mathcal{D}$ for C with $1-\nu(n) > \sqrt{1-\eta_\varepsilon(n)^2}$ for a certain function $\nu(n)$ from $\mathbb{N}$ to $[0,1]$, then there exists an $(\varepsilon(n),\delta(n))$-quantum list-decoder $\mathcal{B}$ for C with oracle access to $\tilde{O}$ such that $\mathcal{B}$ produces a list of size at most*

$$\left\lceil \frac{q(n)}{1-\nu(n)-\sqrt{1-\eta_\varepsilon(n)^2}} \left(\log_e J_{\varepsilon,q,d,M}(n) + \log_e \frac{1}{1-\delta(n)} \right) \right\rceil,$$

where $\eta_\varepsilon(n) = (q(n)/(q(n)-1))\varepsilon(n)$. Moreover, if $\mathcal{D}$ runs in time polynomial in $(n,q(n),\log_2 M(n))$ and $1-\nu(n)-1/g(n) > \sqrt{1-\eta_\varepsilon(n)^2}$ holds for a certain positive-valued function $g(n)$ that is polynomially-bounded in $(n,q(n),\log_2 M(n),1/\varepsilon(n),\log_2(1/(1-\delta(n))))$, then $\mathcal{B}$ runs in time polynomial in $(n,q(n),\log_2 M(n),1/\varepsilon(n),\log_2(1/(1-\delta(n))))$.

By Theorem 4.7 together with Theorem 4.2, we obtain a quantum list-decoder for a given block code family. Now, we give the proof of Theorem 4.7.

Proof of Theorem 4.7. Fix $n \in \mathbb{N}$. Since $\mathcal{D}$ is a $(1-\nu)$-quantum codeword-state decoder for C, for each $x \in \Sigma_n$ and $k \in \mathbb{F}_{q(n)}^{+}$, $\mathcal{D}$ outputs x from the k-shuffled codeword state $|C_x^{(k)}\rangle$ with probability at least $1-\nu(n)$. Let $\tilde{O}$ be any quantumly corrupted codeword for C. Given $\tilde{O}$ as an implicit input, we consider the following algorithm $\mathcal{B}$ that can solve the ε-QLDP for C with probability at least $\delta(n)$. For notational readability, we omit the script "n." Write σ for the value $1-\nu-\sqrt{1-\eta_\varepsilon^2}$. Initially, set $k=1$ in the algorithm $\mathcal{A}$.

Quantum Algorithm $\mathcal{A}$:

(1) Starting with $|0^m\rangle$, run the algorithm $\mathcal{A}$ to obtain the quantum state $|\psi_k\rangle$.

(2) Apply the algorithm $\mathcal{D}$ to the second register of $|\psi_k\rangle$ using an appropriate number of ancilla qubits, say m. We then obtain the state $\mathcal{D}|\psi_k\rangle|0^m\rangle$.

(3) Measure the obtained state and add this measured result to the list of message candidates.

(4) Repeat Steps (1)–(3) $\lceil((1-\sigma)/\sigma)(\log_e J_{\varepsilon,q,d,M}(n) + \log_e(1/(1-\delta)))\rceil$ times.

(5) Repeat Steps (1)–(4) by incrementing k by one at each repetition until $k=q$. Finally, output the list that is produced.

End of the Algorithm

Now, we claim the following. Let $B_\varepsilon^{(k)} = \{x \in \Sigma_n \mid \mathrm{Pre}_{\tilde{O}}(C_x^{(k)}) \geq 1/q+\varepsilon\}$.

Claim 1 *1. With probability at least σ, we can observe x for a certain index k in $\mathbb{F}_q^+$ when measuring the quantum state obtained after Step (2) in the computational basis.*

2. If we proceed Steps (1)–(3) $\lceil((1-\sigma)/\sigma)(\log_e |B_\varepsilon^{(k)}| + \log_e(1/\delta))\rceil$ times for each $k \in \mathbb{F}_q^+$, then we obtain a list that includes all messages in $B_\varepsilon^{(k)}$ with probability at least δ.

Since $|B_\varepsilon^{(k)}| \leq J_{\varepsilon,q,d,M}(n)$, if we repeat Steps (1)-(4) $\lceil((1-\sigma)/\sigma)(\log_e |B_\varepsilon^{(k)}| + \log_e(1/(1-\delta)))\rceil$ times, we obtain a valid list with probability at least δ by the above claim.

Let us prove the Claim 1. The *trace distance* $\|\rho-\sigma\|_{\mathrm{tr}}$ between two quantum states ρ and σ is defined to be $Tr\sqrt{(\rho-\sigma)(\rho-\sigma)^\dagger}$. In particular, for two pure states $|\phi\rangle$ and $|\psi\rangle$, the trace distance between them can be calculated as $\||\phi\rangle\langle\phi| - |\psi\rangle\langle\psi|\|_{\mathrm{tr}} = 2\sqrt{1-|\langle\phi|\psi\rangle|^2}$. For two (probability) distributions D_1 and D_2 over Σ_n, the L_1-norm (or the *total variation distance*) $\|D_1-D_2\|_1$ is defined as $\sum_{x\in\Sigma_n}|D_1(x)-D_2(x)|$.

Proof of Claim 1. (1) Choose $k \in \mathbb{F}_q^+$ and x satisfying that $|\kappa_x^{(k)}| \geq \eta_\varepsilon$. Denote by $p_k(x)$ the probability

of observing x at Step (3) during round k. Our goal is to show that $p_k(x) \geq \sigma$. For simplicity, let $|\phi_{x,k}\rangle = |k\rangle|C_x^{(k)}\rangle|\tau\rangle|0^m\rangle$ and $|\hat{\psi}_k\rangle = |\psi_k\rangle|0^m\rangle$. The trace distance between two pure states $\mathcal{D}|\hat{\psi}_k\rangle$ and $\mathcal{D}|\phi_{x,k}\rangle$ equals

$$\|\mathcal{D}|\phi_{x,k}\rangle\langle\phi_{x,k}|\mathcal{D}^\dagger - \mathcal{D}|\hat{\psi}_k\rangle\langle\hat{\psi}_k|\mathcal{D}^\dagger\|_{\mathrm{tr}} = \||\phi_{x,k}\rangle\langle\phi_{x,k}| - |\hat{\psi}_k\rangle\langle\hat{\psi}_k|\|_{\mathrm{tr}} = 2\sqrt{1-|\langle\phi_{x,k}|\hat{\psi}_k\rangle|^2} = 2\sqrt{1-|\kappa_x^{(k)}|^2}.$$

Let $D_k(y)$ and $\tilde{D}_{x,k}(y)$ be the probabilities of obtaining $y \in \Sigma_n$ by measuring the states $\mathcal{D}|\hat{\psi}_k\rangle$ and $\mathcal{D}|\phi_{x,k}\rangle$, respectively, in the computational basis. Note that $p_k(x)$ equals $\tilde{D}_{x,k}(x)$. If $\tilde{D}_{k,x}(x) \geq 1-\nu$, then obviously $\tilde{D}_{k,x}(x) \geq e$. Next, we deal with the case where $\tilde{D}_{k,x}(x) < 1-\nu$. Since the total variation distance between D_k and $\tilde{D}_{x,k}$ is at most the trace distance between $|k\rangle|C_x^{(k)}\rangle|\tau\rangle$ and $|\psi_k\rangle$, it follows that

$$\|D_k - \tilde{D}_{x,k}\|_1 \leq \|\mathcal{D}|\phi_{x,k}\rangle\langle\phi_{x,k}|\mathcal{D}^\dagger - \mathcal{D}|\hat{\psi}_k\rangle\langle\hat{\psi}_k|\mathcal{D}^\dagger\|_{\mathrm{tr}} = 2\sqrt{1-|\kappa_x^{(k)}|^2}.$$

Moreover, we claim that $\|D_k - \tilde{D}_{x,k}\|_1 \geq 2(1-\nu-\tilde{D}_{x,k}(x))$. This is shown as follows. First, we note that $\|D_k - \tilde{D}_{k,x}\|_1$ is lower-bounded by

$$\begin{aligned}\|D_k - \tilde{D}_{x,k}\|_1 &= |D_k(x) - \tilde{D}_{x,k}(x)| + \sum_{y:y\neq x} |D_k(y) - \tilde{D}_{x,k}(y)| \\ &\geq |D_k(x) - \tilde{D}_{x,k}(x)| + \left|\sum_{y:y\neq x} D_k(y) - \sum_{y:y\neq x} \tilde{D}_{x,k}(y)\right| = 2|D_k(x) - \tilde{D}_{x,k}(x)|\end{aligned}$$

since $\sum_{y\in\Sigma_n} |D_k(y)| = \sum_{y\in\Sigma_n} |\tilde{D}_{x,k}(y)| = 1$. Recall that $D_k(x) \geq 1-\nu$, which implies $D_k(x) \geq \tilde{D}_{k,x}(x)$. We then obtain

$$\|D_k - \tilde{D}_{x,k}\|_1 \geq 2(D_k(x) - \tilde{D}_{k,x}(x)) \geq 2(1-\nu-\tilde{D}_{x,k}(x)).$$

The above two bounds on $\|D_k - \tilde{D}_{k,x}\|_1$ yields the following inequality:

$$1-\nu-\tilde{D}_{x,k}(x) \leq \sqrt{1-|\kappa_x^{(k)}|^2},$$

which immediately implies

$$\tilde{D}_{x,k}(x) \geq 1-\nu-\sqrt{1-|\kappa_x^{(k)}|^2} \geq 1-\nu-\sqrt{1-\eta_\varepsilon^2} = \sigma$$

since $|\kappa_x^{(k)}| \geq \eta_\varepsilon$. Therefore, we conclude that $p_k(x) \geq \sigma$, as requested.

(2) Fix $k \in \mathbb{F}_q^+$ arbitrarily. Assuming that Steps (1)–(4) are repeated t times to create a valid list, we wish to prove that $t \geq ((1-\sigma)/\sigma)(\log_e |B_\varepsilon^{(k)}| + \log_e (1/(1-\delta)))$. Since we obtain $x \in B_\varepsilon^{(k)}$ through these steps with probability at least e, for each fixed $x_0 \in B_\varepsilon^{(k)}$, the probability of obtaining no x_0 within t samples is upper-bounded by $(1-e)^t$. Therefore, with probability at most $|B_\varepsilon^{(k)}|(1-e)^t$, there exists an $x \in B_\varepsilon^{(k)}$ for which t samples does not contain x.

Since the probability of obtaining the desired valid list is at least δ, we demand the condition that $|B_\varepsilon^{(k)}|(1-e)^t \leq 1-\delta$; equivalently,

$$t \log_e \frac{1}{1-\sigma} \geq \log_e |B_\varepsilon^{(k)}| + \log_e \frac{1}{1-\delta},$$

which yields the desired bound

$$t \geq \frac{1-\sigma}{\sigma}\left(\log_e |B_\varepsilon^{(k)}| + \log_e \frac{1}{1-\delta}\right)$$

because $\log_e(1/(1-\sigma))$ is upper-bounded by

$$\log_e \frac{1}{1-\sigma} = \log_e\left(1+\frac{\sigma}{1-\sigma}\right) \leq \frac{\sigma}{1-\sigma}.$$

This completes the proof of the claim. □

Claim 1 guarantees that, if we repeat Steps (1)-(3) $\lceil((1-\sigma)/\sigma)(\log_e |B_\varepsilon^{(k)}| + \log_e (1/(1-\delta)))\rceil$ times, then we can obtain with probability at least δ a valid list of size at most $q(n)\lceil((1-\sigma)/\sigma)(\log_e |B_\varepsilon^{(k)}| + \log_e (1/(1-\delta)))\rceil$. □

5 Nearly Phase-Orthogonal Codes

Theorem 4.7 gives a way to transform a quantum codeword-state decoder into a quantum list-decoder. What types of codes satisfy the premise of the theorem and therefore have quantum list-decoders? Phase-orthogonality of a code family C with message spaces $\{\Sigma_n\}_{n\in\mathbb{N}}$ implies that, for each pair $(n,k) \in \mathbb{N} \times \mathbb{F}^+_{q(n)}$, the set $\{|C_0^{(k)}\rangle, |C_1^{(k)}\rangle, \ldots, |C_{N(n)-1}^{(k)}\rangle\}$, where $N(n) = |\Sigma_n|$, forms an orthonormal basis of an $N(n)$-dimensional Hilbert space for each $k \in \mathbb{F}^+_{q(n)}$. From a practical point of view, however, this requirement for the phase orthogonality is too restrictive to prove the quantum list decodability of a wide range of code families. How can we relax the phase orthogonality of C? A simple solution is to allow C to satisfy the following weaker requirement: $|\langle C_x|C_y\rangle| \leq \delta$ for any distinct pair (x,y).

Definition 5.1 (nearly phase orthogonality) Let $\xi \in [0,1]$. We say that a classical block $(M(n),n)_{q(n)}$-code family C is said to be *ξ-nearly phase-orthogonal* if $|\langle C_x^{(k)}|C_y^{(k)}\rangle| \leq \xi$ for any number $n \in \mathbb{N}$, any element $k \in \mathbb{F}^+_{q(n)}$, and any message pair $x, y \in \Sigma_n$.

Notice that, although any code that is almost "phase orthogonal" is already classically list-decodable (unless there is a decoding time-bound), as we will see later, our quantum list-decoder merits a significantly smaller number of queries (roughly two queries per candidate) than any classical list-decoder.

Unlike the previous sections, we treat every quantum state $|\phi\rangle$ as a column vector. We say that an $(M(n),n)_{q(n)}$-code family C has *full phase-rank* if, for every message length $n \in \mathbb{N}$, the $N(n)$ column vectors $|C_0^{(k)}\rangle, |C_1^{(k)}\rangle, \ldots, |C_{N(n)-1}^{(k)}\rangle$ are linearly independent for each choice of $k \in \mathbb{F}^+_{q(n)}$. Such a code family C satisfies that $M(n) \geq N(n)$ for all $n \in \mathbb{N}$. We show that nearly phase-orthogonal codes that have full phase-rank are indeed quantum list-decodable (ignoring the running time of their quantum list-decoders).

Note that a quantum hardcore $C(x,r)$ requires the condition that $|r| = poly(|x|)$. It thus suffices to consider only $(M(n),n)_{q(n)}$-codes C that satisfy the inequality $M(n) \geq N(n)$, where $N(n)$ is the size of the space of messages of length n.

Theorem 5.2 *Let ε, η be any function from $\mathbb{N}$ to $[0,1]$. Let C be any $(M(n),n)_{q(n)}$-code family of full phase-rank. Assume that there is a function $g(n)$ that is bounded by a certain positive polynomial in $(n, q(n), \log_2 M(n), 1/\varepsilon(n), \log_2(1/(1-\delta(n))))$ and satisfies $1-\eta(n)-1/g(n) > \sqrt{1-(q(n)/(q(n)-1))^2\varepsilon(n)^2}$. If C is η-nearly phase-orthogonal, then C is (ε,δ)-quantum list-decodable with list size polynomial in $(n, q(n), \log_2 M(n), 1/\varepsilon(n), \log_2(1/(1-\delta(n))))$.*

Theorem 5.2 follows from the next lemma using Theorem 4.7. To prove the lemma, we use the notion of pretty-good measurement (known also as square-root measurement or least-squares measurement) [13, 20]. Let E_n denote the n-by-n identity matrix.

Lemma 5.3 *Let η be any function from $\mathbb{N}$ to $[0,1]$. Let C be any $(M(n),n)_{q(n)}$-code family of full phase-rank such that $M(n) \geq N(n)$ for all $n \in \mathbb{N}$. If C is η-nearly phase-orthogonal, then there exists a $(1-\eta)$-quantum codeword-state decoder for C.*

Proof. Fix $n \in \mathbb{N}$ and $k \in \mathbb{F}^+_{q(n)}$. For simplicity, assume that $I_N = [0, N(n)-1]_{\mathbb{Z}}$. Since C has full phase-rank, it follows that $M(n) \geq N(n)$. For readability, we omit the script "n" in the rest of this proof. We wish to construct a unitary operator U whose success probability of obtaining z from $|C_z^{(k)}\rangle$ is at least $1-\eta$ whenever $|\langle C_x^{(k)}|C_y^{(k)}\rangle| \leq \eta$ for any two distinct messages $x, y \in \Sigma_n$.

We want to design U by following an argument of pretty good measurement [13, 20]. Let S be the M-by-N matrix $(|C_0^{(k)}\rangle, |C_1^{(k)}\rangle, \ldots, |C_{N-1}^{(k)}\rangle)$, in which the ith column of S expresses the column vector $|C_i^{(k)}\rangle$. Notice that $S|z\rangle_N = |C_z^{(k)}\rangle$ for each $z \in [0, N-1]_{\mathbb{Z}}$, where $|z\rangle_N$ denotes the N-dimensional unit vector whose zth entry is 1 and 0 elsewhere.

Note that $SS^\dagger$ is an M-by-M matrix and $S^\dagger S$ is an N-by-N matrix. The linear independence of column vectors $|C_0^{(k)}\rangle, |C_1^{(k)}\rangle, \ldots, |C_{N-1}^{(k)}\rangle$ implies that $\mathrm{rank}(S) = N$. Since $S^\dagger S$ and $S^\dagger S$ are Hermitian and positive definite, they also share the same set of positive eigenvalues, say $\{\lambda_0, \ldots, \lambda_{N-1}\}$. Let $\lambda_{\min} = \min\{\lambda_0, \lambda_1, \ldots, \lambda_{N-1}\}$. We claim that $\lambda_{\min}$ is relatively large.

Claim 2 $\lambda_{\min} \geq 1-\eta$.

Proof of Claim 2. Let $G = S^\dagger S$, the N-by-N matrix $(\eta_{i,j})_{i,j}$, where $\eta_{i,j} = \langle C_i^{(k)} | C_j^{(k)} \rangle$. By our assumption, it follows that $|\eta_{ij}| \leq \eta$ for any pairs (i,j). Since G is Hermitian and $\mathrm{rank}(G) = N$, let $G = \sum_{i=0}^{N-1} \lambda_i |\psi_i\rangle\langle\psi_i|$ be a spectral decomposition of G for a certain orthonormal basis $\{|\psi_i\rangle\}_{i \in I_N}$. We then have

$$\min_{\||\psi\rangle\|=1} |\langle\psi|G|\psi\rangle| = \min_{\||\psi\rangle\|=1} \left| \sum_{i=0}^{N-1} \lambda_i |\langle\psi_i|\psi\rangle|^2 \right| = \lambda_{\min}.$$

Note that, for any state $|\phi\rangle$ of the form $\sum_{i \in I_N} \alpha_i |i\rangle$ with complex numbers α_i's, the value $|\langle\psi|G|\psi\rangle|$ equals

$$|\langle\psi|G|\psi\rangle| = \left| 1 + \sum_{i \neq j} \eta_{i,j} \alpha_i^* \alpha_j \right| = \left| 1 - \eta + \sum_i \eta |\alpha_i|^2 + \sum_{i \neq j} \eta_{i,j} \alpha_i^* \alpha_j \right|.$$

Note that an imaginary part of $\sum_{i \neq j} \eta_{i,j} \alpha_i^* \alpha_j$ is zero since for any distinct pair (i,j) we have $\Im(\eta_{i,j} \alpha_i^* \alpha_j + \eta_{j,i} \alpha_j^* \alpha_i) = \Im(\eta_{i,j} \alpha_i^* \alpha_j + \eta_{i,j}^* \alpha_j^* \alpha_i) = \Im(\eta_{i,j} \alpha_i^* \alpha_j + (\eta_{i,j} \alpha_i^* \alpha_j)^*) = 0$. It suffices to consider the case that $\alpha_i \in \mathbb{R}$ and $\eta_{i,j} \in \mathbb{R}$ for all $i,j \in I_N$. For simplicity, write $A^+ = \{(i,j) \in I_N \times I_N \mid i < j, \eta_{ij} \geq 0\}$ and $A^- = \{(i,j) \in I_N \times I_N \mid i < j, \eta_{ij} < 0\}$. Hence, the term $\min_{\||\psi\rangle\|=1} |\langle\psi|G|\psi\rangle|$ equals

$$\min_{\||\psi\rangle\|=1} \left| 1 - \eta + \sum_{(i,j) \in A^+} \left(\eta \alpha_i^2 + 2|\eta_{i,j}| \alpha_i \alpha_j + \eta \alpha_j^2 \right) + \sum_{(i,j) \in A^-} \left(\eta \alpha_i^2 - 2|\eta_{i,j}| \alpha_i \alpha_j + \eta \alpha_j^2 \right) \right|,$$

which is further calculated as

$$\min_{\||\psi\rangle\|=1} \left| 1 - \eta + \sum_{(i,j) \in A^+} \left(|\eta_{i,j}| (\alpha_i + \alpha_j)^2 + (\eta - |\eta_{i,j}|)(\alpha_i^2 + \alpha_j^2) \right) \right.$$
$$\left. + \sum_{(i,j) \in A^-} \left(|\eta_{i,j}| (\alpha_i - \alpha_j)^2 + (\eta - |\eta_{i,j}|)(\alpha_i^2 + \alpha_j^2) \right) \right|.$$

Since $\eta - |\eta_{ij}| \geq 0$ for any pair (i,j), we conclude that the target term $\min_{\||\psi\rangle\|=1} |\langle\psi|G|\psi\rangle|$ is at least $1 - \eta$. We therefore obtain the inequality $\lambda_{\min} \geq 1 - \eta$, as required. □

Let $S = PTQ^\dagger$ be a *singular-value decomposition* (see, e.g., [22]), where P is an M-by-M unitary matrix, Q is an N-by-N unitary matrix, and T is an M-by-N matrix of the form $\begin{pmatrix} T' \\ O \end{pmatrix}$ with the diagonal matrix $T' = \mathrm{diag}(\sqrt{\lambda_0}, \sqrt{\lambda_1}, \ldots, \sqrt{\lambda_{N-1}})$. We therefore have $\langle z|_M US|z\rangle_N = \langle z|_M UPTQ^\dagger|z\rangle_N$ for any $z \in I_N$.

The desired M-by-M matrix U is defined as $U = RP^\dagger$, where the M-by-M matrix R is

$$R = \begin{pmatrix} Q & O \\ O & E_{M-N} \end{pmatrix}.$$

It immediately follows that, for any $z \in I_N$,

$$\begin{aligned} \langle z|_M US|z\rangle_N &= \langle z|_M RTQ^\dagger|z\rangle_N = \langle z|_M \begin{pmatrix} Q & O \\ O & E_{M-N} \end{pmatrix} \begin{pmatrix} T' \\ O \end{pmatrix} Q^\dagger|z\rangle_N \\ &= \langle z|_M \begin{pmatrix} QT'Q^\dagger \\ O \end{pmatrix} |z\rangle_N = \langle z|_N QT'Q^\dagger|z\rangle_N. \end{aligned}$$

The probability of recovering z from $|C_z\rangle$ is therefore lower-bounded by $|\langle z|_N QT'Q^\dagger|z\rangle_N|^2$, which is at least $\lambda_{\min}$. The above claim yields the desired conclusion. □

6 Circulant Codes and a Multiplicative Property

To design quantum codeword-state decoders, Theorem 5.2 gives a general but "non-constructive" method for all nearly phase-orthogonal codes of full phase-rank. Under a certain condition, we can construct quantum codeword-state decoders that run in *polynomial time*. This section presents such conditions using an approximation scheme and also a codeword property.

The (discrete and quantum) Fourier transform is one of the most useful operations in use. A certain type of matrices, known as *circulant matrices*, can be diagonalized by these Fourier transforms. we use these matrices to define new codes, which can be efficiently codeword-state decodable. First, fix a positive integer n and let $L_n = [0, n-1]_{\mathbb{Z}}$. An n-by-n integer matrix $Q = (q_{ij})_{i,j\in[n]}$ is called the *cyclic permutation matrix* if $q_{n,1} = 1$, $q_{i,i+1} = 1$ for any $i \in [n-1]$, and the others are all zeros. Notice that Q^n equals the identity matrix. A *circulant matrix* M is of the form $\sum_{j\in L_n} a_j Q^j$ for certain complex numbers $\{a_j\}_{j\in L_n}$; in other words, the (i,j)-entry of M is $a_{j-i \bmod n}$ for each pair $i, j \in L_n$. Consider the quantum Fourier transform F_n over $\mathbb{Z}_n$. Any circulant matrix $M = \sum_{j\in L_n} a_j Q^j$ can be diagonalized by F_n as follows:

$$F_n^{-1} M F_n = \mathrm{diag}\left(\sum_{j\in L_n} a_j \omega_n^{i\cdot j}\right)_{i\in L_n} = \mathrm{diag}\left(\sum_{j\in L_n} a_j, \sum_{j\in L_n} a_j\omega_n^j, \sum_{j\in L_n} a_j\omega_n^{2j} \ldots, \sum_{j\in L_n} a_j\omega_n^{(n-1)j}\right).$$

Definition 6.1 [circulant code] A classical block code family $C = \{C_i\}_{i\in\mathbb{N}}$ with index sets $\{I_n\}_{n\in\mathbb{N}}$ is said to be *circulant*¶ if, for every message length $n \in \mathbb{N}$, the matrix $M_C^{(n)} = (C_i(j))_{i,j\in I_n}$ is circulant; namely, $M_C^{(n)} = \sum_{i=0}^{|I_n|-1} C_0(i)Q^i$, where Q denotes the M-by-M cyclic permutation matrix.

From the definition of $M_C^{(n)}$ in Definition 6.1, the transposed matrix $(M_C^{(n)})^t = (C_j(i))_{i,j\in I_n}$ can be expressed as $\sum_{j=0}^{|I_n|-1} C_0(|I_n| - j \bmod |I_n|)Q^j$ and therefore it is also a circulant matrix.

Now, let our attention be focused on *shuffled codeword states* of circulant codes. Let C be any $(M(n), n)_{q(n)}$ circulant code with a series $\{I_n\}_{n\in\mathbb{N}}$ of index sets. Consider k-shuffled codeword states $|C_i^{(k)}\rangle$. Conventionally, we treat $|C_i^{(k)}\rangle$ as the column vector $[((1/\sqrt{M})\omega_M^{k\cdot C_i(j)})_{j\in I_n}]^t$ and $\langle C_i^{(k)}|$ as the row vector $((1/\sqrt{M})\omega_M^{-k\cdot C_i(j)})_{j\in I_n}$. We use the notation $M_{k,C}$ to denote the matrix $(|C_0^{(k)}\rangle, \ldots, |C_{M(n)-1}^{(k)}\rangle)$, which equals

$$M_{k,C} = \sum_{j\in I_n} \left(\frac{1}{\sqrt{M(n)}}\, \omega_q^{k\cdot C_0(M(n)-j \bmod M(n))}\right) Q^j$$

and the conjugate transpose of $M_{k,C}$ can be expressed as

$$M_{k,C}^\dagger = \sum_{j\in I_n} \left(\frac{1}{\sqrt{M(n)}}\, \omega_q^{k\cdot C_0(j)}\right) Q^j.$$

Clearly, these matrices are circulant since so are the matrices $(C_i(j))_{i,j\in I_n}$ and $(C_j(i))_{i,j\in I_n}$.

In the following proposition, we prove that, if we can approximate the matrix $F_M M_{k,C} F_M^{-1}$ efficiently, we can construct efficiently a codeword-state decoder for C. In this proposition, we use the notion of *operator norm* $\|A\|$ of a complex square matrix A, defined as $\|A\| = \sup_{|\phi\rangle,|\psi\rangle:\||\phi\rangle\|=\||\psi\rangle\|=1} |\langle\phi|A|\psi\rangle|$.

Proposition 6.2 *Let C be an $(M(n), n)_{q(n)}$ circulant code. Let $k \in \mathbb{F}_{q(n)}^+$, $\delta \in [0,1]$, and let D_k denote $F_{M(n)} M_{k,C}^\dagger F_{M(n)}^{-1}$. For each constant $k \in \mathbb{F}_{q(n)}^+$, let $\tilde{D}_k$ denote a linear operator such that $||\tilde{D}_k - D_k|| \le \delta$, where $||\cdot||$ denotes the operator norm. If $\tilde{D}_k$ is computable in time polynomial in $(n, q(n), \log_2 M(n))$, then C is $(1-\delta)^2$-quantum codeword-state decodable in time polynomial in $(n, q(n), \log_2 M(n))$.*

Proof. Let $k \in \mathbb{F}_{q(n)}^+$. By omitting the script "n," our desired codeword-state decoder U_k that outputs i from $|C_i^{(k)}\rangle$ can be expressed in the form $F_M^{-1}\tilde{D}_k F_M$. Obviously, U_k is a linear operator that can be realized in time polynomial in $(n, q, \log M)$.

Now, we wish to evaluate the the success probability $|\langle i|U_k|C_i^{(k)}\rangle|^2$ of obtaining i by applying U_k to $|C_i^{(k)}\rangle$. For convenience, let $\Delta_k = \tilde{D}_k - D_k$. This Δ_k satisfies the following inequality:

$$|\langle i|F_M^{-1}\Delta_k F_M|C_i^{(k)}\rangle| = |(\langle i|F_M^{-1})\Delta_k(F_M|C_i^{(k)}\rangle)| \le \|\Delta_k\| = \|\tilde{D}_k - D_k\| \le \delta.$$

We then have

$$\begin{aligned} |\langle i|U_k|C_i^{(k)}\rangle| &= \left|\langle i|F_M^{-1}\tilde{D}_k F_M|C_i^{(k)}\rangle|\right| = \left|\langle i|F_M^{-1}(D_k+\Delta_k)F_M|C_i^{(k)}\rangle\right| \\ &\ge \left|\langle i|F_M^{-1}D_k F_M|C_i^{(k)}\rangle\right| - \left|\langle i|F_M^{-1}\Delta_k F_M|C_i^{(k)}\rangle\right|, \end{aligned}$$

¶This notion is different from the codes that have *circulant constructions* (see, e.g., [24]).

which is further bounded by

$$\begin{aligned} |\langle i|U_k|C_i^{(k)}\rangle| &\geq \left|\langle i|M_{k,C}^{\dagger}|C_i^{(k)}\rangle\right| - ||\Delta_k|| \geq \left|\langle i|M_{k,C}^{\dagger}|C_i^{(k)}\rangle\right| - \delta \\ &= \left|\langle C_i^{(k)}|C_i^{(k)}\rangle\right| - \delta = 1-\delta. \end{aligned}$$

Thus, we can obtain i from $|C_i^{(k)}\rangle$ with probability at least $(1-\delta)^2$. Since $\tilde{D}_k$ can be computed in time polynomial in $(n, q, \log M)$, our codeword-state decode also runs in time polynomial in $(n, q, \log M)$. □

Next, we wish to prove that nearly phase-orthogonal circulant codes are quantum codeword-state decodable under a certain ideal condition. We begin with a useful lemma on a family of nearly phase-orthogonal circulant codes.

Lemma 6.3 *Let $\eta \in [0,1]$. Let C be any $(M(n), n)_{q(n)}$ circulant code with a message space $\Sigma^* = \bigcup_{n\in\mathbb{N}} \Sigma_n$ and index sets $\{I_n\}_{n\in\mathbb{N}}$. If C is η-nearly phase-orthogonal, then it follows that, for each element $k \in \mathbb{F}_{q(n)}^{+}$ and each index $r \in I_n$,*

$$\left|\sum_{r\in I_n} \omega_{M(n)}^{-i\cdot r}\omega_{q(n)}^{k\cdot C_0(r)}\right|^2 \geq (1-\eta)M(n).$$

Proof. Let C be any $(M(n), n)_{q(n)}$ circulant code. Fix $n \in \mathbb{N}$ and we often omit the script "n" for readability. For simplicity, we assume that $I_n = [0, M(n)-1]_{\mathbb{Z}}$. Assume that C is η-nearly phase-orthogonal; that is, for any $x, y \in I_n$, $|\langle C_y^{(k)}|C_x^{(k)}\rangle| \leq \eta$ if $y \neq x$.

As noted before, $M_{k,C}$ can be diagonalized by the quantum Fourier transform F_M as follows:

$$F_M^{-1} M_{k,C} F_M = \mathrm{diag}\left(\frac{1}{\sqrt{M}}\sum_{j\in I_n} \omega_M^{-ij}\omega_q^{k\cdot C_0(j)}\right)_{i\in I_n}.$$

Similarly, we obtain the following diagonalization:

$$F_M^{-1} M_{k,C}^{\dagger} F_M = \mathrm{diag}\left(\frac{1}{\sqrt{M}}\sum_{j\in I_n} \omega_M^{ij}\omega_q^{-k\cdot C_0(j)}\right)_{i\in I_n}.$$

From these results, the matrix $M_{k,C}M_{k,C}^{\dagger}$ can be also diagonalized by the quantum Fourier transform as

$$F_M^{-1}(M_{k,C}M_{k,C}^{\dagger})F_M = \mathrm{diag}\left(\left|\frac{1}{\sqrt{M}}\sum_{j\in I_n} \omega_M^{-ij}\omega_q^{k\cdot C_0(j)}\right|^2\right)_{i\in I_n}.$$

Following an argument similar to the proof of Claim 2, each entry value $|(1/\sqrt{M})\sum_{j\in I_n}\omega_M^{-ij}\omega_q^{k\cdot C_0(j)}|^2$ in $F_M^{-1}(M_{k,C}M_{k,C}^{\dagger})F_M$ for $i \in I_n$ is at least $1-\eta$. This yields the lemma. □

Circulant codes are desirable candidates for quantum hardcore functions; however, we do not know if all circulant codes have polynomial-time quantum codeword-state decoder. We need to demand an additional property, called the *multiplicative property*, for circulant codes.

Definition 6.4 (multiplicative property) We say that a classical $(M(n), n))_{q(n)}$-code C enjoys the *multiplicative property* if, for any message length $n \in \mathbb{N}$ and for any two positive indices $i, j \in I_n$, it holds that $C_0(i) + C_0(j) = C_0(i\cdot j) \bmod q(n)$. In particular, we have $C_0(1) = 0$.

For the next proposition, we use the following weak form of "increasing" functions. A function $M(n)$ from $\mathbb{N}$ to $\mathbb{N}$ is said to be *increasing* if, for any number m, there exists another number $n > m$ such that $M(n) > M(m)$.

Proposition 6.5 *Let $M(n)$ be an increasing function. Let η be any function from $\mathbb{N}$ to $[0,1]$. Let C be any $(M(n), n)_{q(n)}$ circulant code that is η-nearly phase-orthogonal. If C is polynomial-time computable and enjoys the multiplicative property, then C is $(1-\eta')$-quantum codeword-state decodable in time polynomial in $(n, q(n), \log_2 M(n))$, where $\eta'(n) = 1-(1-1/M(n))^2(1-\eta(n)) + 6/\sqrt{M(n)}$.*

In comparison, recall that Lemma 5.3 gives a $(1-\eta)$-quantum codeword-state decoder using a non-constructive argument. Proposition 6.5, however, gives a roughly $(1-1/M(n))$ times smaller quantum codeword-state decoder that works in polynomial time. When $M(n)$ is sufficiently large, the proposition gives almost optical quantum codeword-state decoder.

By combining Proposition 6.5 with Theorem 4.2, we obtain the following theorem.

Theorem 6.6 *Let $M(n)$ be an increasing function. Let C be any polynomial-time computable $(M(n),n)_{q(n)}$ circulant code family that enjoys the multiplicative property. Assume that a function $g(n)$ is upper-bounded by a certain positive polynomial in $(n, q(n), 1/\varepsilon(n), \log_2(1/(1-\delta(n))))$ and satisfies the inequality $(1-1/M(n))^2(1-\eta(n))-6/\sqrt{M(n)}-1/g(n) > \sqrt{1-(q(n)/(q(n)-1))^2\varepsilon(n)^2}$. If C is η-nearly phase-orthogonal, then C has an (ε,δ)-quantum list-decoder running in time polynomial in $(n, q(n), \log_2 M(n), 1/\varepsilon(n), \log_2(1/1-\delta(n)))$.*

We show the pending proof of Proposition 6.5.

Proof of Proposition 6.5. We use a quantum algorithm of van Dam, Hallgren, and Ip [28] to obtain the desired bound. Let C be any $(M(n),n)_{q(n)}$-code that is circulant and also η-nearly phase-orthogonal. Assume that C holds the multiplicative property. Let $\{I_n\}_{n\in\mathbb{N}}$ be the index sets of C and write I_n^+ for the set $I_n - \{0\}$. We drop "n" in the rest of this proof. Consider the following quantum algorithm $\mathcal{A}$.

Quantum Algorithm $\mathcal{A}$:

(1) On input $|C_x^{(k)}\rangle$, apply the inverse quantum Fourier transform F_M^{-1}. We then obtain

$$|\phi_1\rangle = \frac{1}{M}\sum_{r,\ell\in I_n} \omega_M^{-r\ell}\omega_q^{C_x^{(k)}(\ell)}|r\rangle = \frac{1}{M}\sum_{r\in I_n}\omega_M^{-rx}\left(\sum_{\ell\in I_n}\omega_M^{-r\ell}\omega_q^{C_0^{(k)}(\ell)}\right)|r\rangle.$$

(2) Transform $|j\rangle$ to $\omega_q^{C_0^{(k)}(j)}|j\rangle$ for each $j\in I_n^+$ and do nothing for $j=0$. This is done in polynomial time (by running an appropriate deterministic (reversible) computation for $C_0(j)$, applying phase encoding, and uncomputing the first deterministic computation) since C is polynomial-time computable. Since $C_0(r)+C_0(\ell)=C_0(r\ell) \bmod M$, we have

$$\begin{aligned}
|\phi_2\rangle &= \frac{1}{M}\sum_{\ell\in I_n}\omega_q^{kC_0(\ell)}|0\rangle + \frac{1}{M}\sum_{r\in I_n^+}\omega_M^{-rx}\left(\omega_q^{C_0^{(k)}(0)+C_0^{(k)}(r)} + \sum_{\ell\in I_n^+}\omega_M^{-r\ell}\omega_q^{C_0^{(k)}(r\ell)}\right)|r\rangle \\
&= \frac{1}{M}\sum_{r\in I_n}\omega_M^{-rx}\left(\sum_{\ell\in I_n^+}\omega_M^{-\ell}\omega_q^{C_0^{(k)}(\ell)}\right)|r\rangle + \frac{1}{M}\omega_q^{C_0^{(k)}(0)}\left(\sum_{r\in I_n^+}\omega_M^{-rx}\omega_q^{C_0^{(k)}(r)}\right)|r\rangle \\
&\qquad + \frac{1}{M}\left[\omega_q^{C_0^{(k)}(0)} + \sum_{\ell\in I_n}\omega_q^{C_0^{(k)}(\ell)}\left(1-\omega_M^{-\ell}\right)\right]|0\rangle.
\end{aligned}$$

(3) Apply the quantum Fourier transform F_M. We then have

$$\begin{aligned}
|\phi_3\rangle &= \left(\frac{1}{\sqrt{M}}\sum_{\ell\in I_n^+}\omega_M^{-\ell}\omega_q^{kC_0(\ell)}\right)|x\rangle + \frac{1}{M}\omega_q^{C_0^{(k)}(0)}\left(\frac{1}{\sqrt{M}}\sum_{r\in I_n^+,s\in I_n}\omega_M^{r(s-x)}\omega_q^{kC_0(r)}|s\rangle\right) \\
&\qquad + \frac{1}{M}\left[\omega_q^{C_0^{(k)}(0)} + \sum_{\ell\in I_n^+}\left(1-\omega_q^{-\ell}\right)\omega_q^{C_0^{(k)}(0)}\right].
\end{aligned}$$

(4) Measure the resulted state. We observe x with probability $|\langle x|\phi_3\rangle|^2$, where

$$\begin{aligned}
\langle x|\phi_3\rangle &= \frac{1}{\sqrt{M}}\left(1-\frac{1}{M}\right)\sum_{\ell\in I_n}\omega_M^{-\ell}\omega_q^{C_0^{(k)}(\ell)} \\
&\qquad + \frac{1}{M\sqrt{M}}\left(1+\omega_q^{kC_0(0)}\right)\sum_{\ell\in I_n^+}\omega_q^{C_0^{(k)}(\ell)} - \frac{1}{\sqrt{M}}\left(1-\frac{2}{M}\right)\omega_q^{C_0^{(k)}(0)}.
\end{aligned}$$

End of the Algorithm

Recall that $M(n)$ is an increasing function. To complete the proof, we wish to show that $|\langle x|\phi_3\rangle| \geq (1-1/M)^2(1-\eta) - 6/\sqrt{M}$ for any sufficiently large $M(n)$. Note that Lemma 6.3 implies that $|(1/\sqrt{M})\sum_{\ell\in I_n}\omega_M^{-\ell}\omega_q^{C_0^{(k)}(\ell)}|^2 \geq 1-\eta$. For any sufficiently large M, it follows that

$$\begin{aligned}
|\langle x|\phi_3\rangle| &\geq \left(1-\frac{1}{M}\right)\left|\frac{1}{\sqrt{M}}\sum_{\ell\in I_n}\omega_M^{-\ell}\omega_q^{C_0^{(k)}(\ell)}\right| - \frac{2}{M\sqrt{M}}\sum_{\ell\in I_n^+}\left|\omega_q^{C_0^{(k)}(\ell)}\right| - \frac{1}{\sqrt{M}}\left(1-\frac{2}{M}\right) \\
&\geq \left(1-\frac{1}{M}\right)\sqrt{1-\eta} - \frac{2}{\sqrt{M}}\left(1-\frac{1}{M}\right) - \frac{1}{\sqrt{M}} \\
&\geq \left(1-\frac{1}{M}\right)\sqrt{1-\eta} - \frac{3}{\sqrt{M}}.
\end{aligned}$$

By squaring $|\langle x|\phi_3\rangle|$, we obtain the desired bound:

$$|\langle x|\phi_3\rangle|^2 \geq \left(1-\frac{1}{M}\right)^2(1-\eta) + \frac{9}{M} - \frac{6\sqrt{1-\eta}}{\sqrt{M}}\left(1-\frac{1}{M}\right) \geq \left(1-\frac{1}{M}\right)^2(1-\eta) - \frac{6}{\sqrt{M}}.$$

This completes the proof. □

7 Finding New Quantum Hardcore Functions

As outlined in Section 3, our goal is to construct a polynomial-time quantum list-decoder for a target quantum hardcore candidate. How can we prove the quantum hardcore property of such a candidate? We quickly review our arguments of the previous sections. In Theorem 3.4, we show that quantum list-decodability yields the quantum hardcore property. With the help of Theorem 4.2, Theorem 4.7 further relates quantum codeword-state decodability to quantum list-decodability. The remaining task is to prove the quantum codeword-state decodability of a quantum hardcore candidate. The following lemma summarizes our argument.

Lemma 7.1 *Let s be any negligible[‖] function mapping $\mathbb{N}$ to the unit real interval $[0,1]$. Let C be any $(M(n),n)_{q(n)}$-code family with $\log_2 M(n) \in n^{O(1)}$ and $q(n) \in n^{O(1)}$. Assume that there exists a polynomial-time $(1-s(n))$-quantum codeword-state decoder for C. For any noticeable function ε, there exist a noticeable function δ and a polynomial-time (ε,δ)-quantum list-decoder C. Hence, C satisfies the quantum hardcore property.*

Proof. Let s be any negligible function and let $\mathcal{D}$ be a polynomial-time $(1-s(n))$-quantum codeword-state decoder for an $(M(n),n)_{q(n)}$-code family C. Let ε be any noticeable function. By the definition of noticeability, there is an appropriate positive polynomial p' such that $\varepsilon(n) \geq 1/p'(n)$ for any sufficiently large $n \in \mathbb{N}$. Define $\delta(n) = 1-2^{-n}$ for any $n \in \mathbb{N}^+$. Note that $\log_2(1/(1-\delta(n))) = \log_2 2^n = n$. To apply Theorem 4.7, we need to show that $\Gamma(n) = (1-s(n)) - \sqrt{1-(q(n)/(q(n)-1))^2\varepsilon(n)^2}$ is a noticeable function (in n), because the functions $\log_2 M(n)$, $q(n)$, $1/\varepsilon(n)$, and $\log_2(1/(1-\delta(n)))$ are all polynomially bounded in n. Fix a sufficiently large number n in $\mathbb{N}$. Using the inequality $\sqrt{1-x} \geq 1-x/2$, we obtain

$$\Gamma(n) \geq 1-s(n)-\sqrt{1-\varepsilon(n)^2} \geq 1-s(n)-\left(1-\frac{1}{2p'(n)^2}\right) = \frac{1}{2p'(n)^2} - s(n).$$

Since s is a negligible function, clearly Γ is a noticeable function. □

For circulant codes, we obtain the following general result, which is a direct consequence of Theorem 6.6.

Lemma 7.2 *Let $M(n)$ be an increasing function. Let C be any polynomial-time computable $(M(n),n)_{q(n)}$ circulant code family $C(x,r)$ that enjoys the multiplicative property. Assume that $\log_2 M(n) \in n^{O(1)}$ and $q(n) \in n^{O(1)}$. Let η be any negligible function. If C is η-nearly phase-orthogonal, then C satisfies the quantum hardcore property.*

[‖] A function μ mapping $\mathbb{N}$ to $[0,1]$ is said to be *negligible* if, for any positive polynomial p, there exists a number $n_0 \in \mathbb{N}$ such that $\mu(n) \leq 1/p(n)$ for all $n \geq n_0$.

Proof. This proof is similar to that of Lemma 7.1. Let C be any polynomial-time computable $(M(n), n)_{q(n)}$ circulant code family that satisfies the multiplicative property. Assume that C is η-nearly phase-orthogonal for a certain negligible function η. To appeal to Theorem 3.4, we need to prove that, for arbitrary noticeable function ε, there exists another noticeable function δ such that $\Gamma(n) = (1-1/M(n))^2(1-\eta(n))-6/\sqrt{M(n)}-\sqrt{1-(q(n)/(q(n)-1))^2\varepsilon(n)^2}$ is also a noticeable function. Take a positive polynomial $\hat{p}$ such that $\varepsilon(n) \geq 1/\hat{p}(n)$ for all $n \in \mathbb{N}$. Note that $\sqrt{1-x} \leq 1-x/2$ for any $x \in [0,1]$. We estimate the value Γ as follows.

$$\begin{aligned}
\Gamma(n) &\geq \left(1-\frac{1}{M(n)}\right)(1-\eta(n)) - \frac{6}{\sqrt{M(n)}} - \sqrt{1-\frac{1}{\hat{p}(n)^2}} \\
&\geq 1-\eta(n)-\frac{1}{M(n)}+\frac{\eta(n)}{M(n)}-\frac{6}{\sqrt{M(n)}}-\left(1-\frac{1}{2\hat{p}(n)^2}\right) \\
&\geq \frac{1}{2\hat{p}(n)^2}-\eta-\frac{7}{\sqrt{M(n)}}.
\end{aligned}$$

Since $M(n)$ is exponentially large and $\eta(n)$ is negligible, we conclude that $\Gamma(n)$ is lower-bounded by $1/p'(n)$ for a certain positive polynomial p' for any sufficiently large number $n \in \mathbb{N}$. □

Now, let us present three new quantum hardcore functions, two of which are unknown to be classically hardcores. These new quantum hardcores are (i) q-ary Hadamard codes, (ii) shifted Legendre symbol codes, and (iii) pairwise equality codes. Goldreich and Levin [15] showed that binary Hadamard codes are classical hardcores and q-ary Hadamard codes are later shown to be classical hardcores by Goldreich, Rubinfeld, and Sudan [16]. We explain these quantum hardcores as codes and give polynomial-time quantum list-decoding algorithms for them. With help of Theorem 4.7, however, we need to build only their quantum codeword-state decoders instead of quantum list-decoders.

Proposition 7.3 *Let $p(n)$ and $q(n)$ be any two functions mapping $\mathbb{N}$ to the prime numbers. There exists a polynomial-time quantum list-decoding algorithm for each of the following codes:*

1. *The $q(n)$-ary* Hadamard code $\mathrm{HAD}^{(q)}$ *with $q(n) \in n^{O(1)}$, whose codeword $\mathrm{HAD}^{(q)}_x$ is defined as $\mathrm{HAD}^{(q)}_x(r) = \sum_{i=0}^{2^n-1} x_i \cdot r_i \bmod q(n)$. The minimal distance $d(\mathrm{HAD}^{(q)})$ is $(1-1/q(n))q(n)$.*
2. *The* shifted Legendre symbol code SLS^p, *which is a $(p(n), n)_2$-code with $n = \lceil \log p(n) \rceil$ and odd prime $p(n)$, whose codeword SLS^p_x is defined by the Legendre symbol** as $\mathrm{SLS}^p_x(r) = 1$ if $\left(\frac{x+r}{p(n)}\right) = -1$, and $\mathrm{SLS}^p_x(r) = 0$ otherwise.*
3. *The* pairwise equality code PEQ *for even numbers $n \in \mathbb{N}$, which is a $(2^n, n)_2$-code, whose codeword is $\mathrm{PEQ}_x(r) = \bigoplus_{i=0}^{n/2} \mathrm{EQ}(x_{2i}x_{2i+1}, r_{2i}r_{2i+1})$, where* EQ *denotes the equality predicate (i.e., $\mathrm{EQ}(x,y) = 1$ if $x = y$ and 0 otherwise) and $\oplus$ is the bitwise XOR.*

By applying Theorem 3.4, we can prove the quantum hardcore property of all the codes given in Proposition 7.3.

Theorem 7.4 *The functions $\mathrm{HAD}^{(q)}$, SLS^p, and* PEQ *are all quantum hardcore functions for any quantum one-way function f' of the form $f'(x,r) = (f(x), r)$ for any x and any r with $|r| = s(|x|)$, where f is an arbitrary quantum one-way function and s is a polynomial.*

Earlier, Damgård [10] introduced the so-called *Legendre generator*, which takes input $(p(n), x)$ and produces a $q(n)$-bit sequence whose rth bit equals $\mathrm{SLS}^p_x(r)$ for every index $r \in \mathbb{F}_{q(n)}$, where q is a fixed polynomial. He asked whether his generator possesses the classical hardcore property (which is also listed as an open problem in [17].) Our result, Proposition 7.3(2), proves the "quantum" hardcore property of Damgård's generator for any quantum one-way function.

Finally, we give the proof of Proposition 7.3.

Proof of Proposition 7.3. By Lemma 7.1, it suffices to provide a polynomial-time $(1-1/s(n))$-quantum codeword-state decoder for each of the given codewords in Proposition 7.3, where s is a certain negligible function. Now, fix $n \in \mathbb{N}$ and omit "n" for readability.

**For any odd prime p, let $\left(\frac{x}{p}\right) = 0$ if $p|x$, $\left(\frac{x}{p}\right) = 1$ if $p \not| x$ and x is a quadratic residue modulo p, and $\left(\frac{x}{p}\right) = -1$ otherwise.

(1) The simple case $q = 2$ was implicitly proven by Adcock and Cleve [1] and also by Bernstein and Vazirani [6]. Consider the general case $q \geq 2$. We want to show that $\mathrm{HAD}^{(q)}$ is 1-quantum codeword-state decodable. By Lemma 7.1, this implies the quantum list-decodability of $\mathrm{HAD}^{(q)}$. Let $|\mathrm{HAD}^{(q)}\rangle$ be a codeword state. To recover x from the codeword state $|\mathrm{HAD}^{(q)}\rangle$, we note that

$$F_q|x\rangle = \frac{1}{\sqrt{q}} \sum_{r\in\mathbb{F}_q} \omega_q^{x\cdot r}|r\rangle = \frac{1}{\sqrt{q}} \sum_{r\in\mathbb{F}_q} \omega_q^{\mathrm{HAD}_x^{(q)}(r)}|r\rangle = |\mathrm{HAD}_x^{(q)}\rangle.$$

Hence, apply the inverse of F_q to $|\mathrm{HAD}_x^{(q)}\rangle$ and we immediately obtain $|x\rangle$ with probability 1.

(2) We consider a new code C defined as follows: let $C_i(j)$ be $\mathrm{SLS}_{-i}^p(j)$ (using "$-i$" instead of "i") for each pair $i, j \in I_n$. Since C is a circulant code, we hereafter consider its associated matrix $M_{2,C}^{\dagger} = \sum_{j\in I_n} \left((1/\sqrt{p})\omega_2^{C_0(j)}\right) Q^j$.

To obtain a quantum codeword-state decoder for C_i, we use Proposition 6.2. First, we define a useful constant c_p as follows: $c_p = 1$ if $p \equiv 1 \bmod 4$, and $c_p = \iota$ (i.e., the unit of imaginary numbers) if $p \equiv 3 \bmod 4$. This constant c_p satisfies the following equation (see e.g. [9]):

$$(*) \qquad \frac{1}{\sqrt{p}} \sum_{j\in\mathbb{F}_p} \left(\frac{j}{p}\right) \omega_p^{a\cdot j} = c_p \left(\frac{a}{p}\right)$$

for any number $a \in [0, p-1]_{\mathbb{Z}}$. Let $D_2 = F_p^{-1} M_{2,C}^{\dagger} F_p$, which equals

$$D_2 = \mathrm{diag}\left(\frac{1}{\sqrt{p}} \sum_{j\in I_n} \omega_p^{ij} \omega_2^{-C_0(j)}\right)_{i\in I_n} = \mathrm{diag}\left(\frac{1}{\sqrt{p}} + \frac{1}{\sqrt{p}} \sum_{j\in I_n} \left(\frac{j}{p}\right) \omega_p^{ij}\right)_{i\in I_n}$$

because $\omega_2^{-C_0(0)} = 1$ and $\omega_2^{-C_0(i)} = \left(\frac{i}{p}\right)$ for any number $i \in \mathbb{F}_p^+$. By (*), we have

$$D_2 = \mathrm{diag}\left(\frac{1}{\sqrt{p}} + c_p\left(\frac{i}{p}\right)\right)_{i\in I_n} = \mathrm{diag}\left(\frac{1}{\sqrt{p}}, \frac{1}{\sqrt{p}} + c_p\omega_2^{-C_0(1)}, \ldots, \frac{1}{\sqrt{p}} + c_p\omega_2^{-C_0(p-1)}\right).$$

We define our desired linear operator $\tilde{D}_2$ as $\tilde{D}_2 = \mathrm{diag}\left(0, c_p\omega_2^{-C_0(1)}, \ldots, c_p\omega_2^{-C_0(p-1)}\right)$. This definition makes the operator norm $\|D_2 - \tilde{D}_2\|$ equal

$$\|D_2 - \tilde{D}_2\| = \left\|\mathrm{diag}\left(\frac{1}{\sqrt{p}}, \ldots, \frac{1}{\sqrt{p}}\right)\right\| = \frac{1}{\sqrt{p}}.$$

How can we realize this $\tilde{D}_2$? The operator $\tilde{D}_2$ can be realized by the following polynomial-time algorithm.

> On input $|i\rangle|0\rangle$, where $i \in \mathbb{F}_q$, compute $c_p|i\rangle|C_0(i)\rangle$ in a reversible fashion. Apply the phase-shift transform that changes $|i\rangle|a\rangle$ to $\omega_2^{-a}|i\rangle|a\rangle$. Uncompute $|C_0(i)\rangle$ in the last register and we obtain $c_p\omega_2^{-C_0(i)}|i\rangle|0\rangle$. Finally, when $i = 0$, we reject the input.

Therefore, Proposition 6.2 gives a $(1 - 1/\sqrt{p})^2$-quantum codeword-state decoder for C that runs in time polynomial in n. Since $(1 - 1/\sqrt{p})^2 \geq 1 - 2/\sqrt{p}$, it suffices to define $s(n) = 2/\sqrt{p}$.

To list-decode SLS, since $\mathrm{SLS}_i^p(j) = C_{-i}(j)$, we first find $-i$ from the codeword $C_{-i}(\cdot)$ and then output i. This procedure gives rise to a quantum list-decoder for SLS.

(3) We want to prove that PEQ has a polynomial-time 1-quantum codeword-state decoder. We first observe the following key equation:

$$\begin{aligned} |\mathrm{PEQ}_x(r)\rangle &= \frac{1}{\sqrt{2^n}} \sum_{r=0} (-1)^{\mathrm{PEQ}_x(r)}|r\rangle \\ &= \frac{1}{\sqrt{4}} \sum_{r_1,r_2} (-1)^{\mathrm{EQ}(x_1x_2,r_1r_2)}|r_1r_2\rangle \otimes \cdots \otimes \frac{1}{\sqrt{4}} \sum_{r_{n-1},r_n} (-1)^{\mathrm{EQ}(x_{n-1}x_n,r_{n-1}r_n)}|r_{n-1}r_n\rangle. \end{aligned}$$

Let us consider the following unitary transform H_C, which we call the *circulant Hadamard transform*:

$$H_C =_{def} \begin{pmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix} = F_4^{-1} \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} F_4,$$

where F_4 is the quantum Fourier transform over $\mathbb{F}_4$. Since H_C satisfies the equality

$$H_C\left(\frac{1}{\sqrt{4}}\sum_{r_i,r_{i+1}}(-1)^{\mathrm{EQ}(x_ix_{i+1},r_ir_{i+1})}|r_ir_{i+1}\rangle\right)=|x_ix_{i+1}\rangle,$$

we can obtain $|x_1x_2\rangle\otimes\cdots\otimes|x_{n-1}x_n\rangle$ from the codeword state $|\mathrm{PEQ}_x\rangle$ by applying $U=H_C^{\otimes n/2}$. From this quantum state, we can easily obtain x with probability 1. □

References

[1] M. Adcock and R. Cleve. A quantum Goldreich-Levin theorem with cryptographic applications. In *Proc. of the 19th International Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science, Vol.2285, Springer, pp.323–334, 2002.

[2] A. Akavia, S. Goldwasser, and S. Safra. Proving hard-core predicates using list decoding. *Theoretical Computer Science*, 378(1), 46–53, 2007.

[3] A. Ambainis, K. Iwama, A. Kawachi, R. H. Putra, and S. Yamashita. Robust quantum algorithms for oracle identification. *Quantum Information Processing*, 4(5), 355–386, 2005.

[4] A. Atici and R. Servedio. Improved bounds on quantum learning algorithms. To appear in *Quantum Information Processing*. Available also at http://arxiv.org/abs/quant-ph/0411140.

[5] A. Barg and S. Zhou. A quantum decoding algorithm for the simplex code. In *Proc. of Allerton Conference on Communication, Control and Computing*, 1998. Available at http://citeseer.ist.psu.edu/barg98quantum.html.

[6] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5): 1411–1473, 1997.

[7] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Comput.*, 13: 850–864, 1984.

[8] H. Buhrman, I. Newman, H. Röhrig, and R. de Wolf. Robust quantum algorithms and polynomials. In *Proc. of the 20th International Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science, Vol.3404, pp.593–604, 2003.

[9] R. Crandall and C. Pomerance. *Prime Numbers: A Computational Perspective*, Springer-Verlag, 2001.

[10] I. B. Damgård. On the randomness of Legendre and Jacobi sequences. In *Proc. of the 8th Annual International Cryptology Conference*, Lecture Notes in Computer Science, Vol.403, pp.163–172, 1988.

[11] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6): 644–654, 1976.

[12] D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. In *Proc. Roy. Soc. London*, A, Vol.439, pp.553–558, 1992.

[13] Y. C. Eldar and G. D. Forney, Jr. On quantum detection and the square-root measurement. *IEEE Trans. Inform. Theory*, 47(3):858–872, 2001.

[14] O. Goldreich. *Foundations of Cryptography: Basic Tools*, Cambridge University Press, 2001.

[15] O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In *Proc. of the 21st Annual ACM Symposium on Theory of Computing*, pp.25–32, 1989.

[16] O .Goldreich, Rubinfeld, and M. Sudan. Learning polynomials with queries: the highly noisy case. In *Proc. of the 36th Annual Symposium on Foundations of Computer Science*, pp.294–303, 1995.

[17] M. I. González Vasco and M. Näslund. A survey of hard core functions. In *Proc. of Workshop on Cryptography and Computational Number Theory*, Birkhauser, pp.227–256, 2001.

[18] L. K. Grover. Quantum Mechanics helps in searching for a needle in a haystack. Phys. Rev. Lett. 79(2):325–328, 1997.

[19] V. Guruswami and M. Sudan. Extensions to the Johnson bound. Manuscript, 2000. Available at http://theory.csail.mit.edu/~madhu/.

[20] P. Hausladen and W. K. Wootters. A 'pretty good' measurement for distinguishing quantum states. *J. Mod. Opt.*, 41:2385–2390, 1994.

[21] T. Holenstein, U. M. Maurer, and J. Sjödin. Complete classification of bilinear hard-core functions. In *Proc. of the 24th Annual International Cryptology Conference*, Lecture Notes in Computer Science, Vol.3152, pp.73–91, 2004.

[22] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[23] P. Høyer, M. Mosca, and R. de Wolf. Quantum search on bounded-error inputs. In *Proc. of the 33rd International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science, Vol.2719, pp.291–299, 2003.

[24] W. C. Huffman and V. Pless. *Fundamentals of Error-Correcting Codes.* Cambridge University Press, 2003.

[25] A. Kawachi and T. Yamakami. Quantum hardcore functions by complexity-theoretical quantum list decoding. In *Proc. of the 33rd International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science, Vol.4052 (Part II), pp.216–227, 2006.

[26] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

[27] M. Sudan. List decoding: Algorithms and applications. *SIGACT News*, 31(1): 16–27, 2000.

[28] W. van Dam, S. Hallgren, and L. Ip. Quantum algorithms for some hidden shift problems. *SIAM J. Comput.* 36(3), 763–778, 2006.

[29] T. Yamakami. A foundation of programming a multi-tape quantum Turing machine. In *Proc. of the 24th International Symposium on Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, Springer-Verlag, Vol.1672, pp.430-441, 1999.

[30] A. C. Yao. Quantum circuit complexity. In *Proc. of the 34th Annual Symposium on Foundations of Computer Science*, pp.352–361, 1997.

Appendix: Proof of Lemma 3.2

For the proof of Lemma 3.2, we need to elaborate the brief description given in Section 3 on an interpretation of *presence*. For readability, we fix n and omit this n (for example, we write "q" instead of "$q(n)$") in the following proof. Let v be any quantumly corrupted word that $\tilde{O}$ represents. We view this v as the real vector in the qN dimensional real space defined as follows: let $v[r]$, the rth block of v, be $(|\alpha_{r,1}|^2, |\alpha_{r,2}|^2, \ldots, |\alpha_{r,q}|^2)$ if $\tilde{O}|r\rangle|s\rangle|t\rangle = \sum_{z\in[q]} \alpha_{r,z}|r\rangle|s\oplus z\rangle|t\oplus\phi_{r,z}\rangle$ for certain s, t. Let $\{C_1, C_2, \ldots, C_m\}$ be the set of all codewords that lie "close" to the given quantumly corrupted word. For each C_i, we define c_i to be the corresponding vector defined as follows: let $c_i[r]$, the rth block of c_i, consists of zeros and one 1 at the zth component if $C_i(r)$ outputs z. Let $\mathcal{K}_i = \{x \mid \sum_z x_{r,z} = 0\}$ for each i and set $\mathcal{K} = \bigcap_{i=1}^q \mathcal{K}_i$. Note that $dim(\mathcal{K}) = N(q-1)$. Take any distinct pair (i,j) (i.e., $i \neq j$). We further introduce a new parameter $\beta \in [0,1]$ and define $w = \beta \cdot v + \frac{1-\beta}{q} \cdot \vec{1}$, where $\vec{1}$ is the vector of all 1s. Note that $\langle\vec{1}|\vec{1}\rangle = Nq$. Note also that the (Hamming) distance $d(C_i, C_j)$ between codes C_i and C_j is lower-bounded by d. Moreover, we have $\langle c_i|v\rangle = \sum_r |\alpha_{r,C_i(r)}|^2 = N \cdot \mathrm{Pre}_{\tilde{O}}(C_i)$, where $\langle v|w\rangle$ denotes the standard inner product of two vectors v and w.

The first Upper Bound. We calculate the value $\langle c_i|w\rangle$, $\langle w|w\rangle$, and $\langle c_i|c_j\rangle$ as follows:

$$\begin{aligned}
\langle c_i|w\rangle &= \beta\langle c_i|v\rangle + \frac{1-\beta}{q}\langle c_i|\vec{1}\rangle = \beta N \mathrm{Pre}_{\tilde{O}}(C_i) + \frac{1-\beta}{q}N \geq \beta N\left(\frac{1}{q}+\epsilon\right) + \frac{N(1-\beta)}{q}.\\
\langle w|w\rangle &= \beta^2\langle v|v\rangle + \frac{2\beta(1-\beta)}{q}\langle v|\vec{1}\rangle + \frac{(1-\beta)^2}{q2}\langle\vec{1}|\vec{1}\rangle = N\beta^2 + \frac{2N\beta(1-\beta)}{q} + \frac{N(1-\beta)^2}{q}.\\
\langle c_i|c_j\rangle &= N - d(C_i, C_j) \geq N - d.
\end{aligned}$$

Hence, we obtain:

$$\begin{aligned}
\langle c_i - w|c_j - w\rangle &= \langle c_i|c_j\rangle + \langle w|w\rangle - \langle c_i|w\rangle - \langle c_j|w\rangle\\
&\leq N - d + N\beta^2 + \frac{2N\beta(1-\beta)}{q} + \frac{N(1-\beta)^2}{q} - 2N\left[\left(\frac{1}{q}+\epsilon\right)\beta + \frac{1-\beta}{q}\right]\\
&= N\left(1-\frac{1}{q}\right)\left[\beta^2 - \frac{2q\epsilon}{q-1}\beta + 1\right] - d.
\end{aligned}$$

We assume that $d = \left(1-\frac{1}{q}\right)(1-\delta)N$ for a certain $\delta \in [0,1]$. It thus follows that:

$$\begin{aligned}
\langle c_i - w|c_j - w\rangle &\leq N\left(1-\frac{1}{q}\right)\left[\beta^2 - \frac{2q\epsilon}{q-1}\beta + 1\right] - \left(1-\frac{1}{q}\right)(1-\delta)N\\
&= N\left(1-\frac{1}{q}\right)\left[\beta^2 - \frac{2q\epsilon}{q-1}\beta + \delta\right].
\end{aligned}$$

To make $\langle c_i - w | c_j - w \rangle < 0$, we need to satisfy that $\beta^2 - \frac{2q\epsilon}{q-1}\beta + \delta < 0$, which is equivalent to $\epsilon > \frac{1}{2}\left(1-\frac{1}{q}\right)\left(\beta + \frac{\delta}{\beta}\right)$. To minimize the value $\beta + \frac{\delta}{\beta}$, it suffices to take $\beta = \sqrt{\delta}$. By replacing β by $\sqrt{\delta}$, we obtain $\epsilon > \sqrt{\delta}\left(1-\frac{1}{q}\right)$. Since $\delta = 1 - \frac{d}{N}\left(1+\frac{1}{q-1}\right)$, we obtain the bound $\epsilon > \left(1-\frac{1}{q}\right)\sqrt{1-\frac{d}{N}\left(1+\frac{1}{q-1}\right)}$.

Now, let $\hat{c}_i = c_i - v$ for each i and let $\hat{w}$ be the projection of w onto $\mathcal{K}$. Since $c_i - w \in \mathcal{K}$ and $\epsilon > 1 - 1/q$, we have

$$\langle \hat{c}_i | \hat{w} \rangle = \langle \hat{c}_i | w \rangle \geq \frac{N\beta}{q}\left[q\epsilon - \beta(q-1)\right].$$

Since $\beta = \sqrt{\delta}$, we have

$$\langle \hat{c}_i | \hat{w} \rangle > \frac{N\sqrt{\delta}}{q}\left[q\sqrt{\delta}\left(1-\frac{1}{q}\right) - \sqrt{\delta}(q-1)\right] = 0.$$

As shown in [1], this implies that $m \leq dim(\mathcal{K}) = N(q-1)$.

The Second Upper Bound. We show the second upper bound. Recall that $\langle c_i - w | c_j - w \rangle \leq N\left(1-\frac{1}{q}\right)\left[\beta^2 - \frac{2q\epsilon}{q-1}\beta + \delta\right]$. We choose $\beta = \frac{q\epsilon}{q-1}$ so that $\beta^2 - \frac{2q\epsilon}{q-1} = -\beta^2$. If $\delta < \beta^2 = \left(\frac{q\epsilon}{q-1}\right)^2$, then clearly, we have $\langle c_i - w | c_j - w \rangle \leq N\left(1-\frac{1}{q}\right)(\delta - \beta^2) < 0$. Note that the condition $\delta < \left(\frac{q\epsilon}{q-1}\right)^2$ is equivalent to $\epsilon > \sqrt{\delta}\left(1-\frac{1}{q}\right)$. Since $\delta = 1 - \frac{qd}{N(q-1)}$, we obtain that $\epsilon > \left(1-\frac{1}{q}\right)\sqrt{1-\frac{d}{N}\left(1+\frac{1}{q-1}\right)}$ as before. We also have

$$\|c_i - w\|^2 = \langle c_i - w | c_i - w \rangle \leq N\left(1-\frac{1}{q}\right)\left[\beta^2 - \frac{2q\epsilon}{q-1}\beta + 1\right] = N\left(1-\frac{1}{q}\right)\left[1-\beta^2\right].$$

By normalizing $c_i - w$, we write $u_i = \frac{c_i - w}{\|c_i - w\|}$. Hence, we have

$$\langle u_i | u_j \rangle = \frac{\langle c_i - w | c_j - w \rangle}{\|c_i - w\| \cdot \|c_j - w\|} \leq \frac{\delta - \beta^2}{1-\beta^2} = -\frac{\beta^2 - \delta}{1-\beta^2}.$$

As shown in [19], we obtain the bound:

$$m \leq 1 + \frac{1-\beta^2}{\beta^2 - \delta} = \frac{1-\delta}{\beta^2 - \delta} = \frac{\frac{qd}{N(q-1)}}{\frac{qd}{N(q-1)} - 1 + \left(\frac{q\epsilon}{q-1}\right)^2} = \frac{d\left(1-\frac{1}{q}\right)}{d\left(1-\frac{1}{q}\right) + N\epsilon^2 - N\left(1-\frac{1}{q}\right)^2}.$$

The Equality Case. Assume that $\epsilon(n) = \left(1-\frac{1}{q(n)}\right)\sqrt{1-\frac{d(n)}{N(n)}\left(1+\frac{1}{q(n)-1}\right)}$, which is equivalent to $\epsilon = \sqrt{\delta}\left(1-\frac{1}{q}\right)$. We want to show that $m \leq 2N(n)(q(n)-1) - 1$. Recall that $\langle c_i - w | c_j - w \rangle \leq N\left(1-\frac{1}{q}\right)\left[\beta^2 - \frac{2q\epsilon}{q-1}\beta + \delta\right]$. Taking $\beta = \frac{q\epsilon}{q-1}$ $(= \sqrt{\delta})$, we immediately obtain $\langle c_i - w | c_j - w \rangle \leq 0$. Note that

$$\begin{aligned}
\langle c_i - w | w \rangle &= \langle c_i | w \rangle - \langle w | w \rangle \\
&\geq \beta N\left(\frac{1}{q} + \epsilon\right) + \frac{N(1-\beta)}{q} - N\beta^2 - \frac{2N\beta(1-\beta)}{q} - \frac{N(1-\beta)^2}{q} \\
&= \frac{N}{q}\left[\beta(1+q\epsilon) + (1-\beta) - q\beta^2 - \beta(1-\beta) - (1-\beta)^2\right].
\end{aligned}$$

Replacing β and ϵ by the appropriate terms using δ, we have

$$\langle c_i - w | w \rangle \geq \frac{N}{q}\left[\left(1 + q \cdot \frac{\sqrt{\delta}(q-1)}{q}\right)\sqrt{\delta} - q\delta\right] = \frac{N}{q}\left[\sqrt{\delta} - \delta\right] \geq 0$$

because $0 \leq \delta \leq 1$. Hence, by [19], we obtain that $m = 2N(q-1) - 1$.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de/