Quantum Algorithms inspired by Regev's reduction

André Chailloux

Inria de Paris

January 24 2026 - QIP'26 Tutorial Part III, Riga

q-ary Linear Codes

All of this framework can be easily extended to the q-ary setting.

Definition

A q-ary linear code $\mathcal{C}$ of dimension k and length n is characterized by a full rank generating matrix $\mathbf{G} \in \mathbb{F}_q^{k \times n}$, and we write

$$\mathcal{C} = \{\mathbf{sG} : \mathbf{s} \in \mathbb{F}_q^k\}.$$

Each code $\mathcal{C}$ has an associated $\mathcal{C}^\perp = \{\mathbf{y} \in \mathbb{F}_q^n : \forall \mathbf{c} \in \mathcal{C},\ \mathbf{y} \cdot \mathbf{c} = 0\}$.

$$\mathcal{C}^\perp = \{\mathbf{y} \in \mathbb{F}_q^n : \mathbf{G}\mathbf{y}^\intercal = \mathbf{0}\}.$$

Definition

Let $f : \mathbb{F}_q^n :\to \mathbb{C}$. We define

$$\widehat{f}(\mathbf{x}) = \frac{1}{\sqrt{q^n}} \sum_{\mathbf{y}\in\mathbb{F}_q^n} \chi_{\mathbf{x}}(\mathbf{y})|\mathbf{y}\rangle,$$

with $\chi_{\mathbf{x}}(\mathbf{y}) = \Pi_{i=1}^n \chi_{x_i}(y_i)$ and the χ_{x_i} are the **characters of** $\mathbb{F}_q$.

In the case q is prime, we have

$$\widehat{f}(\mathbf{x}) = \frac{1}{\sqrt{q^n}} \sum_{\mathbf{y}\in\mathbb{F}_q^n} \omega_q^{\mathbf{x}\cdot\mathbf{y}} f(\mathbf{y})|\mathbf{y}\rangle \quad \text{with } \omega_q = e^{\frac{2i\pi}{q}}.$$

Quantum Fourier Transform:

$$\sum_{\mathbf{x}\in\mathbb{F}_q^n} f(x)|\mathbf{x}\rangle \xrightarrow{QFT_q^n} \sum_{\mathbf{x}\in\mathbb{F}_q^n} \widehat{f}(\mathbf{x})|\mathbf{x}\rangle.$$

Reduction in the q-ary setting

The reduction works in exactly the same way

$$\boxed{\sum_{\substack{\mathbf{c}\in\mathcal{C}\\ \mathbf{e}\in\mathbb{F}_q^n}} \chi_{\mathbf{c}}(\mathbf{z}) f(\mathbf{e})|\mathbf{c}+\mathbf{e}\rangle|\mathbf{c}\rangle} \xrightarrow[\text{Solver}]{\text{QDP}} \boxed{\sum_{\substack{\mathbf{c}\in\mathcal{C}\\ \mathbf{e}\in\mathbb{F}_q^n}} \chi_{\mathbf{c}}(\mathbf{z}) f(\mathbf{e})|\mathbf{c}+\mathbf{e}\rangle} \xrightarrow{QFT_q^n} \boxed{\sum_{\mathbf{y}\in\mathcal{C}^\perp} \widehat{f}(\mathbf{y}-\mathbf{z})|\mathbf{y}\rangle}.$$

Measure the last state : CCP solver.

Theorem (Regev's reduction, v3)

Let $T \subseteq \mathbb{F}_q^n$. Let $f : \mathbb{F}_q^n \to \mathbb{C}$ s.t. $Supp(\widehat{f}) \subseteq T$ and $\|f\|_2 = 1$.

Algorithm for $QDP(\mathcal{C}, f)$ with failure $\varepsilon \Rightarrow$

Quantum algorithm for $\mathrm{CCP}(\mathcal{C}^\perp, T)$ with failure $\leq \varepsilon + \eta + 2\sqrt{\varepsilon(1-\varepsilon)\eta}$

The Fourier Transform of a q-ary Bernoulli is also a q-ary Bernoulli. If

$$\sum_{\mathbf{e}\in\mathbb{F}_q^n} f(\mathbf{e})|\mathbf{e}\rangle = \left(\sqrt{1-t}|0\rangle + \sum_{j\neq 0}\sqrt{\frac{t}{q-1}}|j\rangle\right)^{\otimes n}.$$

Then

$$\sum_{\mathbf{e}\in\mathbb{F}_q^n} \widehat{f}(\mathbf{e})|\mathbf{e}\rangle = \left(\sqrt{1-t^*}|0\rangle + \sum_{j\neq 0}\sqrt{\frac{t^*}{q-1}}|j\rangle\right)^{\otimes n},$$

with

$$t^* = \frac{\left(\sqrt{(q-1)(1-t)} - \sqrt{t}\right)^2}{q}.$$

CLZ21 algorithm

[ChenLiuZhandry21], original attempt of this approach for quantum algorithms.

We write $\mathbb{F}_q = \{-\lfloor \frac{q-1}{2} \rfloor, \ldots, \lfloor \frac{q}{2} \rfloor\}$. Take

$$T = \{\mathbf{y} \in \mathbb{F}_q^n : \|\mathbf{y}\|_\infty \leq r\} \quad r = \frac{q}{2} - cst.$$

Infinity norm constraint: For $\mathbf{y} = y_1, \ldots, y_n$, this means each $y_i \in \{-r, \ldots, r\}$.

Take f s.t. $\widehat{f} = \mathbb{1}_T$. Instantiation with random q-ary codes

$$\boxed{\mathrm{DP}(\mathcal{C}, f) \text{ Algo}} \Rightarrow \boxed{\mathrm{CCP}(\mathcal{C}^\perp, T) \text{ Quantum Algo.}}$$

- With $Supp(\widehat{f}) \subseteq \{-r, \ldots, r\}$, impossibility to do unambiguous state discrimination.
- Instead: Each c_i is or erased (as in the erasure channel), or we have the guarantee that $c_i \in S \subseteq \mathbb{F}_q$, with $|S|$ constant.
- Such errors can be efficiently decoded for some parameters: **Arora-Ge algorithm**. Requires $n = O(k^{|S|})$.
- The corresponding $\mathrm{CCP}(\mathcal{C}^{\perp}, T)$ has recently been dequantized [ImranIvanyos23,KothariO'DonnellWu25].
- **No quantum advantage**

- We have our reductions. For any choice code $\mathcal{C}$ and function $f : \mathbb{F}_q^n \to \mathbb{C}$ with $Supp(\widehat{f}) \approx T$.

$$\boxed{\mathrm{QDP}(\mathcal{C}, f) \text{ Quantum Algo}} \Rightarrow \boxed{\mathrm{CCP}(C^{\perp}, T) \text{ Quantum Algo.}}$$

$$\boxed{\mathrm{DP}(\mathcal{C}, |f|^2) \text{ Algo}} \Rightarrow \boxed{\mathrm{CCP}(\mathcal{C}^{\perp}, T) \text{ Quantum Algo.}}$$

- Maybe, if we take **structured codes**, we can find instances where even $\mathrm{DP}(\mathcal{C}, |f|^2)$ is easy for classical computers and we believe $\mathrm{CCP}(\mathcal{C}^{\perp}, T)$ is hard.
- **Road for quantum advantage.**

Decoded Quantum Interferometry

In [Jordan et al. 24], they mainly look at

- **Low Density Parity Check (LDPC) codes**. We have efficient heuristic algorithm (Belief Propagation) to decode LDPC codes. The resulting problem in the dual can be phrased as a Max LinSAT problem.
- **Reed-Solomon codes**. We know efficient decoders. The resulting problem in the dual can be phrased as an approximate polynomial interpolation problem.
- Emphasis on solving $\mathrm{CCP}(\mathcal{C}^{\perp}, T)$ problems that are **natural optimization problems.**

LDPC codes: binary case

Definition

An LDPC code $\mathcal{C}$ is a code that has a sparse parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{(n-k)\times n}$. Typically a constant number of 1s on each row.

Recall that $\mathbf{H}$ is also a generating matrix od $\mathcal{C}^\perp$

We have good heuristic classical decoders for LDPC codes. What happens when we plug this in Regev's reduction?

$$\boxed{\mathrm{DP}(\mathcal{C}, t) \text{ Algo}} \Rightarrow \boxed{\mathrm{CCP}(\mathcal{C}^\perp, t^*) \text{ Quantum Algo.}}$$

I'm solving a $\mathrm{CCP}(\mathcal{C}^\perp, t^*)$ problem where $\mathbf{H} \in \mathbb{F}_2^{(n-k)\times n}$ is a sparse generating matrix of $\mathcal{C}^\perp$.

Definition (Close Codeword Problem $\mathrm{CCP}(\mathcal{C}^\perp, t^*)$)

Goal: For $\mathbf{z} \leftarrow \mathbb{F}_2^n$, find $\mathbf{s} = (s_1, \ldots, s_{n-k}) \in \mathbb{F}_2^{n-k}$ s.t. $|\mathbf{sH} - \mathbf{z}| \leq t^* n$.

Rewriting as a sparse system

$$\begin{aligned} s_1 h_{11} + s_2 h_{21} + \cdots + s_{n-k} h_{(n-k)1} &= z_1 \\ s_1 h_{12} + s_2 h_{22} + \cdots + s_{n-k} h_{(n-k)2} &= z_2 \\ \vdots \qquad\qquad \vdots \qquad\qquad &\vdots \\ s_1 h_{1n} + s_2 h_{2n} + \cdots + s_{n-k} h_{(n-k)n} &= z_n. \end{aligned}$$

$(n-k)$ variables and n equations. I can fail for $t^* n$ equations!

We solve **Max XorSAT.** Can be extended to q-ary **Max LinSAT**.

We instantiate

$$\boxed{\mathrm{DP}(\mathcal{C}, t) \text{ Algo}} \Rightarrow \boxed{\mathrm{CCP}(\mathcal{C}^{\perp}, t^*) \text{ Quantum Algo.}}$$

with LPDC codes to solve a Max XorSAT problem.

- Is this problem **hard for classical computers?**
- The parameters obtained fall a little short.
- Current status: Unclear.

Next example: instantiate with Reed-Solomon codes. Clearer quantum advantage.

Reed-Solomon codes

Definition (Full support Reed-Solomon codes)

Let $\mathbb{F}_q = \{-\lfloor \frac{q-1}{2} \rfloor, \ldots, \lfloor \frac{q}{2} \rfloor\}$. The full support Reed-Solomon code $\mathbf{RS}_k$ of dimension k is the linear code

$$\mathbf{RS}_k \stackrel{\triangle}{=} \{(P(\alpha))_{\alpha \in \mathbb{F}_q} \mid P \in \mathbb{F}_q[X], \deg P < k\} \subseteq \mathbb{F}_q^q.$$

Proposition

We have $\mathbf{RS}_k^\perp = \mathbf{RS}_{q-k}$.

Different decoders for Reed-Solomon codes

Efficient decoders for $\mathbf{RS}_k$.

- The **Berlekamp-Welch decoder** has no failures but decodes to half the minimal distance.
- The **Guruswami-Sudan decoder** can decode more errors but has some failures. This decoder is improved by the **Koetter-Vardy decoder**.

These decoders are efficient **in the injective regime** (DP) but do not carry over to the **surjective regime** (CCP)

We instantiate as follows with $Supp(\widehat{f}) \approx T$.

$$\boxed{\mathrm{DP}(\mathbf{RS}_k, |f|^2) \text{ Algo}} \Rightarrow \boxed{\mathrm{CCP}(\mathbf{RS}_{n-k}, T) \text{ Quantum Algo.}}$$

We write $\mathbb{F}_q = \{-\lfloor \frac{q-1}{2} \rfloor, \ldots, \lfloor \frac{q}{2} \rfloor\}$. One interesting case

$$T = \{\mathbf{y} \in \mathbb{F}_q^n : \|\mathbf{y}\|_\infty \leq r\}.$$

We actually consider f s.t. $\widehat{f} = \mathbb{1}_T$.

$$T = \{\mathbf{x} \in \mathbb{F}_q^n : \|\mathbf{x}\|_\infty \leq r\}.$$

Definition (Close Codeword Problem $\mathrm{CCP}(\mathbf{RS}_{q-k}, T)$)

Sample: $\mathbf{z} \leftarrow \mathbb{F}_q^q$.
Goal: Find $\mathbf{c} \in \mathbf{RS}_{q-k}, \|\mathbf{c} - \mathbf{z}\|_\infty \leq r$.

Recall that each $\mathbf{c} \in \mathbf{RS}_{q-k}$ can be written

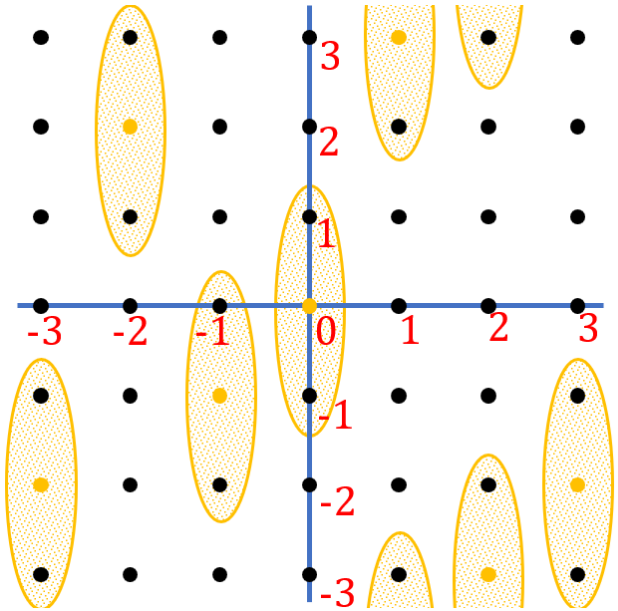
$$\mathbf{c} = (P(\alpha))_{\alpha \in \mathbb{F}_q},$$

for some polynomial $P \in F_q[X]$ and $deg(P) < q - k$.

Solving $\mathrm{CCP}(\mathbf{RS}_{q-k}, T)$ means finding a low degree polynomial P s.t. each $P(\alpha)$ is close to the corresponding z_α for $\alpha \in \mathbb{F}_q$.

Approximate polynomial interpolation

Example: we take $q = 7, r = 1$ and
$\mathbf{z} = (z_{-3}, z_{-2}, \ldots, z_3) = (-2, 2, -1, 0, 3, -3, -2)$.



Finding an point in $\mathbf{y} \in \mathbf{RS}_{q-k}$ such that $\|\mathbf{y} - \mathbf{z}\|_\infty \leq 1$ means finding a polynomial $P \in \mathbb{F}_q[X]$ of degree $< q - k$ that goes through all the orange regions.

Results

Results

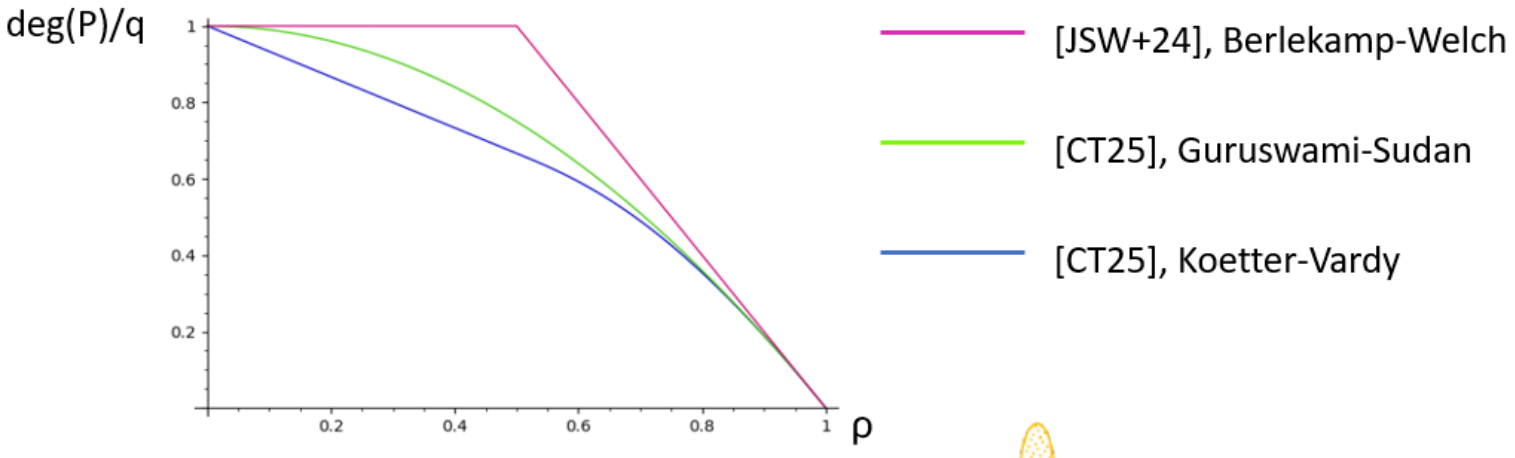


Figure 1: Degree ratio deg(P)/q as a function of $\rho = \frac{\ell}{q}$

Strong exponential speedups over best classical algorithms (which are exponential for any constant $deg(P)/q < 1$).

We instantiated for f with $\widehat{f} = \mathbb{1}_T$.

$$\boxed{\mathrm{DP}(\mathbf{RS}_k, |f|^2) \text{ Algo}} \Rightarrow \boxed{\mathrm{CCP}(\mathbf{RS}_{n-k}, T) \text{ Quantum Algo.}}$$

with

$$T = \{\mathbf{y} \in \mathbb{F}_q^n : \|\mathbf{y}\|_\infty \leq r\}.$$

We can generalize this approach to:

- $T = S_1 \times \ldots S_n$ for random $S_i \subseteq \mathbb{F}_q$.
- Threshold settings (DQI): only satisfy a fraction of constraints.

Moreover:

- We believe these problems are hard for classical computers.
- [YamakawaZhandry22]: There is a choice of T for which we have provable quantum advantage in the Random Oracle Model

CCP and Random Oracle Model

- For each $i \in \mathbb{F}_q$, we consider a random function $h_i : \mathbb{F}_q \to \{0,1\}$ and define

$$S_i = \{\alpha \in \mathbb{F}_q : h_i(\alpha) = 0\}.$$

- **Random Oracle Model:** We are only allowed black box queries to each h_i.
- Rephrasing of $\mathrm{CCP}(\mathbf{RS}_{q-k}, T)$ with $T = S_1 \times \cdots \times S_n$: find $\mathbf{y} = (y_1, \ldots, y_q) \in \mathbf{RS}_{q-k}$ s.t.

$$\forall i \in \mathbb{F}_q,\ h_i(y_i - z_i) = 0, \quad \text{for random } \mathbf{z} \in \mathbb{F}_q^q.$$

- We have random functions $h_i : \mathbb{F}_q \to \{0,1\}$ with black box access.
- We want to find $(y_1, \ldots, y_q) \in \mathbf{RS}_{q-k}$ such that $\forall i \in \mathbb{F}_q,\ h_i(y_i - z_i) = 0$.

Combination of a local constraint $\forall i,\ h_i(y_i - z_i) = 0$ and a global constraint $\mathbf{y} \in \mathbf{RS}_{q-k}$.

- We can solve this problem quantumly if we have a quantum black box access to the function h_i. This is necessary for constructing the corresponding $\sum_{\mathbf{e} \in \mathbb{F}_q^q} f(\mathbf{e})|\mathbf{e}\rangle$.
- Can we prove this problem is hard if we only have **classical access** to the h_i? **No!** Each h_i has only q possible inputs: possible to query them all ($q = n$).

- Idea to circumvent this: more global constraints.
- Use random $h_i : \mathbb{F}_q^{\sqrt{q}} \to \{0,1\}$ for $i \in \{1,\dots,\sqrt{q}\}$ (q square).
- Goal: Find $\mathbf{y} = (y_1,\dots,y_q) \in \mathbf{RS}_{q-k}$ and

$$h_1\Big((y_1,\dots,y_{\sqrt{q}}) - (z_1,\dots,z_{\sqrt{q}})\Big) = 0$$
$$h_2\Big((y_{\sqrt{q}+1},\dots,y_{2\sqrt{q}}) - (z_{\sqrt{q}+1},\dots,z_{2\sqrt{q}})\Big) = 0$$
$$\vdots \qquad\qquad \vdots$$
$$h_{\sqrt{q}}\Big((y_{q-\sqrt{q}+1},\dots,y_q) - (z_{q-\sqrt{q}+1},\dots,z_q)\Big) = 0$$

[YamakawaZhandry22]: There is some choice of k such that:

- **Easy for quantum computers**
- **Hard of classical computers in the ROM**

They proved it even for $\mathbf{z} = \mathbf{0}$ instead of random $\mathbf{z}$.

Recap on the strategies of the main step

$$|\Psi_{\mathbf{c}}\rangle|0\rangle \to |\Psi_{\mathbf{c}}\rangle|\mathbf{c}\rangle, \quad \text{with } |\psi_{\mathbf{c}}\rangle = \sum_{\mathbf{e}\in\mathbb{F}_q^n} f(\mathbf{e})|\mathbf{c}+\mathbf{e}\rangle$$

1. **Use directly a classical decoder** $\mathcal{A}_{Dec} : \mathbf{c}+\mathbf{e} \to \mathbf{c}$ that works for this code $\mathcal{C}$ and choice of function f.
2. **First go through reshaping** the error function f into another error function $\widetilde{f}$ and then apply a good decoder for this $\widetilde{f}$.

Can we use the fact that the errors are in quantum superposition only for reshaping the error function? **No! Already in 2017:**

Belief propagation decoding of quantum channels by passing quantum messages

Joseph M. Renes
Institute for Theoretical Physics, ETH Zurich, 8093 Zürich, Switzerland

The belief propagation algorithm is a powerful tool in a wide range of disciplines from statistical physics to machine learning to computational biology, and is ubiquitous in decoding classical error-correcting codes. The algorithm works by passing messages between nodes of the factor graph associated with the code and enables efficient decoding of the channel, in some cases even up to the Shannon capacity. Here we construct the first belief propagation algorithm which passes *quantum messages* on the factor graph and is capable of decoding the classical-quantum channel with pure state outputs. This gives explicit decoding circuits whose number of gates is quadratic in the code length. We also show that this decoder can be modified to work with polar codes for the pure state channel and as part of a decoder for transmitting quantum information over the amplitude damping channel. These represent the first explicit capacity-achieving decoders for non-Pauli channels.

This line of work (then [PiveteauRenes2021,2025] for instance) show very good algorithms for constructing $|\psi_{\mathbf{c}}\rangle|0\rangle \to |\psi_{\mathbf{c}}\rangle|\mathbf{c}\rangle$ in the case of LDPC codes and Turbo codes. **It's a purely quantum algorithm**.

Recap of main existing results

NB	Structure	Alphabet Size	Norm	Quantum Advantage?
1	Random	2	Hamming	Unknown [CT24]
2	Random	any	$\|\cdot\|_\infty$	Unknown, [CLZ22] Dequantized by [COW25]
3	LDPC	2	Hamming	Unknown, but closer [JSW+24]
4	Reed–Solomon	$\mathrm{poly}(n)$	$\|\cdot\|_\infty$	Yes, SIS_∞ for RS codes [JSW+24, CT25]
5	Folded Reed–Solomon	$\exp(n)$	random	Yes, proven in the ROM [YZ24]

- Numbers 1-2 : required for attacks on post-quantum systems.
- Numbers 3-4-5 : useful for quantum advantage. Several other variants of 4.
- Number 5 gives a lot of hope: this family of quantum algorithms gives provable quantum advantage!

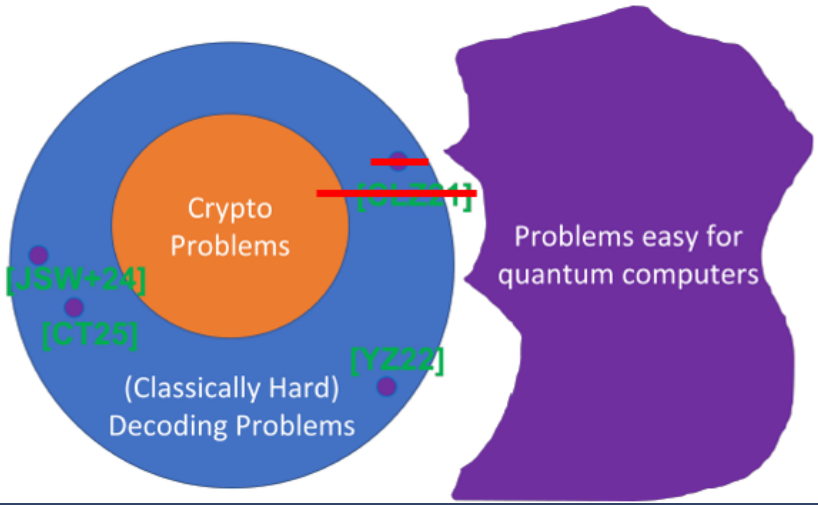
Perspectives

Different possible scenarios. For quantum advantage

- Quantum advantage limited to very specific problems.
- Quantum advantage for a large range of optimization problems.

For post-quantum cryptography

- No implications for post-quantum cryptography.
- Break some of the weakest post-quantum proposals.
- Full break of post-quantum cryptography.



Thank you for your attention.