Statistical Zero Knowledge Problems for the Quantum Support Vector Machine

Saleh Naghdi, Casey Myers, and Michelle Simmons

In a seminal result, Liu *et al.* recently showed that the Quantum Kernel Estimation Support Vector Machine (`QKE-SVM`) provably outperforms all classical models on a classification task that reduces to the famous discrete logarithm problem [6]. This raises a question: are there other such learning problems? To this end, we present a new classification task reducible to order-finding that is able to be learned efficiently by the `QKE-SVM`. We additionally outline the conditions under which learning this task would be classically intractable. Our approach relies on a three-part framework that is centered on the class of Statistical Zero Knowledge (SZK) problems whose instances have the curious property of being separable by a hyperplane within some quantum Hilbert space. We show that if a quantum computer can efficiently map into this space for a given SZK-based classification task, then that task is efficiently learnable by a `QKE-SVM`. Such an efficient mapping typically requires efficiently uncomputing the index of a one-way function in a procedure known as index erasure. Notably, our order-finding classification task is able to bypass the requirement for index erasure as it belongs to a version of SZK where the verifier has access to a quantum computer.

1 Introduction

Quantum machine learning (QML) emerged in 2009 following the work of Lloyd *et al.* on an algorithm for quantum principal component analysis which performed exponentially faster than the best known classical models [7]. Their breakthrough algorithm imminently led to the proliferation of QML models that promised quantum speedup, many of which adopted similar matrix-exponentiation techniques employed in the original paper [2, 5, 8, 9].

Despite their promises for quantum speedup, there remains a large caveat to all of these QML models: in order to be successfully implemented, they share a dependency on quantum RAM, which, analogous to classical RAM, enables classical data to be efficiently loaded in superposition into a quantum computer [3]. Relying on quantum RAM, however, is problematic as the architecture is yet to be experimentally realised due to the difficulty of storing quantum states for long times without immediate decoherence.

Until a feasible implementation of quantum RAM is discovered, the promised quantum speedups behind the breakthrough QML findings remain as conjectures [?]. Worse still, even if QRAM were to be realised, it has been shown that a classical computer with similar data-loading capabilities as QRAM can host quantum-inspired classical algorithms that provide just as much exponential speedup as their quantum counterparts [?]. This raises the suspicion that the source of these speedups lies not in the clever design of the QML algorithm, but rather in its dependence on an unrealistic input model.

To address the impracticality of quantum RAM, the Quantum Kernel Estimation Support Vector Machine (`QKE-SVM`) was introduced in 2018 [4]. Unlike earlier attempts, the `QKE-SVM` does not rely on QRAM and can be implemented on intermediate-scale quantum computers. Whereas QRAM-based QML models require data to be queried in superposition, the `QKE-SVM` relaxes this data-loading assumption to more general quantum feature maps, where the data merely parameterises an arbitrary unitary circuit.

It was not until 2021 when Liu *et al.* showcased a specific classification task on which the `QKE-SVM` admits a provable quantum speedup over all classical counterparts [6]. In their paper, the authors design a classification task that is equivalent to the famous discrete logarithm problem (DLP). Given that the DLP is widely believed to be classically intractable, by showcasing that the QKE-SVM is able to learn their task efficiently they also theoretically demonstrate quantum advantage in QML. Liu *et al.*'s DLP-based classification task raises an important question: are there any other classification tasks that can enjoy a similar quantum speedup? We answer this question in the affirmative.

In this paper, we provide a comprehensive review of Liu *et al.*'s classification task and examine the unique mathematical properties that make it classically intractable while efficiently learnable by a QKE-SVM. We encapsulate these mathematical properties in a three-part framework that revolves around promise problems belonging to the class of Statistical Zero Knowledge (SZK) problems. To demonstrate the efficacy of this framework in constructing classification tasks with desirable properties, we provide two illustrative examples. The first example, serving as a counterexample, involves a classification task within the SZK complexity class and is based on the famous quadratic residuosity problem. This task fails to meet all the criteria in our proposed framework because of its dependence on a subroutine called index erasure. We show that the requirement for index erasure can be dismissed if the SZK class is extended to allow the verifier to have access to a quantum computer. Under this modification, we design another classification task that reduces to order finding. Alongside Liu *et al.*'s findings, this classification task stands as the only other documented instance where a potentially classically intractable task can be efficiently learned exclusively by a QKE-SVM.

2 Review of Liu *et al.*'s classification task based on the discrete logarithm problem

As the only instance of end-to-end quantum machine learning achieving quantum advantage in the literature, the work of Liu *et al.* deserves a closer exploration. Liu *et al.*'s classification task is equivalent to the famous discrete logarithm problem (DLP) which is widely conjectured to be classically intractable. In this section, will first state the discrete logarithm problem and the complexity classes to which it belongs. Then, we will review Liu *et al.*'s classification task in the language of a promise problem Π^{DLP} which naturally gives rise to a pair of disjoint classes labelled as either YES or NO. We give a quantum feature map that maps these instances to a Hilbert space in which there exists a max-margin hyperplane separating them. Finally, we present the algorithm for efficiently implementing the quantum feature map which crucially makes use of the quantum algorithm for efficiently solving the DLP. In what follows, it should be noted that the discrete logarithm problem (DLP) is not to be confused with the discrete logarithm promise problem (Π^{DLP}) even if computationally they are equivalent to each other.

2.1 The discrete logarithm problem

With origins in number theory, the discrete logarithm problem (DLP) has served as the basis for many well-known cryptography protocols that are conjectured to be classically intractable [**?**].

Let p be a prime number and $\mathbb{Z}_p^* = \{1, ..., p-1\}$ the integers under multiplication modulo p. If g is a generator of $\mathbb{Z}_p^*$, then given $y \in \mathbb{Z}_p^*$, the DLP asks for $x \in \mathbb{Z}_p^*$ such that $y = g^x$. In analogy to the real-valued logarithm, such a quantity x is called the discrete logarithm $\log_g(y) = x$. To summarise,

Discrete Logarithm Problem

Input:	A prime number p, a generator g of the set $\mathbb{Z}_p^* = \{1, 2, \cdots, p-1\}$, and an element $y \in \mathbb{Z}_p^*$
Question:	What is $\log_g(y)$?

It is widely conjectured that there is no classical algorithm capable of efficiently evaluating $\log_g(y)$ for large p whereas a bounded-error polynomial-time quantum algorithm for the DLP does exist which is based on Shor's algorithm. This places the DLP within BQP. In fact, an exact quantum algorithm has been shown to exist for the DLP, furthermore placing it within EQP [**? ?**]. The DLP also happens to belong to the class of statistical zero knowledge problems SZK which as we will discuss later plays a significant role in its ability to be learned by a support vector machine.

2.2 The discrete logarithm promise problem Π_{DLP}

The discrete logarithm promise problem Π^{DLP} is defined as a tuple of YES and NO instances,

$$\begin{aligned}\Pi^{DLP} &= (\Pi_Y^{DLP}, \Pi_N^{DLP}) \qquad (1)\\ \Pi_Y^{DLP} &= \{y \ : \ h_{\text{DLP},s}(y) = +1\}\\ \Pi_N^{DLP} &= \{y \ : \ h_{\text{DLP},s}(y) = -1\} \ ,\end{aligned}$$

which are designated as such according to the labeling rule

$$h_{\text{DLP},s}(y) = \begin{cases} +1, & \text{if } \log_g(y) \in [s, s + \frac{p-3}{2}] \\ -1, & \text{else.}\end{cases} \qquad (2)$$

with $y, s \in \mathbb{Z}_p^*$. Note that the interval in the definition sweeps exactly half of the elements in $\mathbb{Z}_p^*$.[1]

Before we discuss the above labeling rule in more depth, a few comments about the distinction between promise problem, classification task and dataset are in order. A promise problem Π (as defined for a special case in Equation 2) can be considered to generate a data space $\mathcal{X} = (\Pi_Y \times \{\text{YES}\}) \cup (\Pi_N \times \{\text{NO}\})$ of data points and their respective label. A dataset $D \subset \mathcal{X}$ is a sample obtained from the data space. The classification task induced by this promise problem entails predicting the label of an unseen point $x \in \Pi_Y \cup \Pi_N$ provided a dataset $\mathcal{X}$. In this way, a promise problem defines a corresponding classification task and dataset.

To illustrate what the labelling rule in 2 corresponds to, it is instructive to first imagine the $p-1$ many elements of $\mathbb{Z}_p^*$ as being distributed evenly along a circle. While in this representation the YES and NO class labels seem to be distributed randomly (Figure 1a) – and indeed, they are to a classical classifier – mapping each point to the position of its discrete logarithm $y \mapsto \log_g(y)$ reveals the underlying classification rule. In this logarithm space $\log_g(\mathbb{Z}_p^*)$ the two labels are separated by a diameter line intercepting the point s, with YES (NO) points lying clockwise (anticlockwise) from s (Figure 1b).

The existence of this diameter partition should already allude to there being a hyperplane in some feature space that differentiates between the two class labels. The next section shows not only one such feature map, but importantly one that is quantum.

2.3 A quantum feature map for separating Π^{DLP}

Let $n = \lceil \log_2(p) \rceil$ be the minimum number of bits required to represent prime number p and define (y, g, s) as before. We will show that the quantum feature space defined by the map

$$|y\rangle \mapsto \sum_{i=0}^{2^k-1} |y \cdot g^i \mod p\rangle := |C_{y,k}\rangle \, . \qquad (3)$$

for a choice of a constant $k \leq n$ admits a hyperplane that separates the data points based on their classes. The data points y are represented as feature states $|C_{y,k}\rangle$ in this feature space.

Consider the state

$$|\phi_s\rangle = \sum_{i=0}^{(p-1)/2-1} |g^s \cdot g^i \mod p\rangle \ , \qquad (4)$$

which is a superposition of all $y \in \Pi_Y^{DLP}$. In Figure 1b, $|\phi_s\rangle$ is depicted as the blue arc sweeping a semicircle across all the YES instances which are depicted in green. The inner product of this state and a given feature state $|C_{y,k}\rangle$ largely depends on the label of y[2]:

$$\begin{aligned}|\langle \phi_s | C_{y,k}\rangle| &= \sum_{i=0}^{2^k-1} \sum_{j=0}^{(p-1)/2-1} \langle y \cdot g^i | g^s \cdot g^j\rangle\\ &= \sum_{i=0}^{2^k-1} \sum_{j=0}^{(p-1)/2-1} \delta_{g^{i+\log_g(y)}, g^{j+s}}\\ &= \sum_{i=0}^{2^k-1} \sum_{j=0}^{(p-1)/2-1} \delta_{i+\log_g(y), j+s}\end{aligned}$$

Note that this expression merely corresponds to the length of the overlap between the arcs swept by $|\phi_s\rangle$ and feature state $|C_{y,k}\rangle$:

$$\begin{aligned}&= |[\log_g(y), \log_g(y) + 2^k - 1] \cap [s, s + \frac{p-1}{2} - 1]|\\ &= |[\max(\log_g(y), s), \min(\log_g(y) + 2^k - 1, s + \frac{p-1}{2} - 1)]|\end{aligned}$$

[1] The addition in the interval is a slight abuse of notation as addition is not a native operation of $\mathbb{Z}_p^*$, so we have (p-1) + 1 = 1

[2] We have temporarily omitted normalisation factors in this derivation for clarity

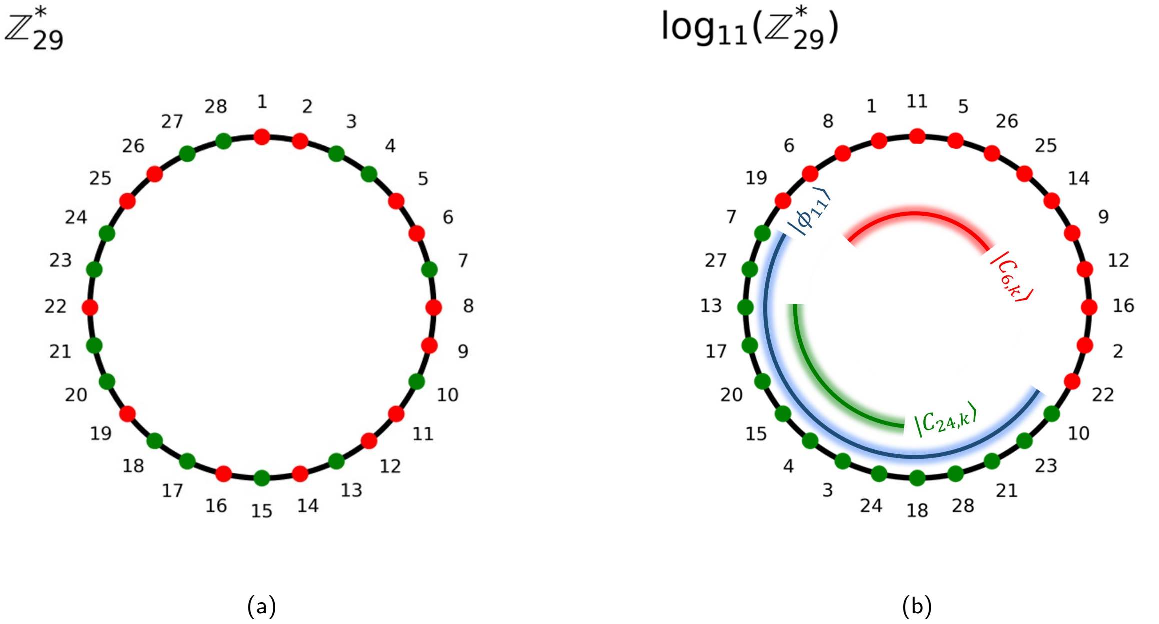


Figure 1: **Use of the quantum feature map 3 to separate the yes/no (green/red) instances of Π^{OFP} for the special case of $(p, g, s, k) = (29, 11, 10, 3)$.** In 1a, the labeled points lying along the circumference are integers $y \in \mathbb{Z}^*_{29}$ with labels assigned according to the labeling rule $h_{\text{DLP},10}$. For example, the data point $y = 28$ is labelled green because $log_{11}(28) = 14 \in= [s, s + (p-1)/2 - 1] = [11, 24] \implies h_{11}(x) = +1$. These points form our data space. In 1b, the data points have been mapped to the position of their discrete logarithm action. For example, since $y = 28 \mapsto log_{11}(28) = 14$, $x = 29$ is mapped to the position of 14. The concentric arcs represent the feature states $|C_{y,k}\rangle$ for $y = 24 \in \Pi_Y$ (green) and $y = 6 \in \Pi_N$ (red), as well as the normal to the hyperplane $|\phi_1 1\rangle$ defined in 4 which a quantum computer is efficiently able to map the data points to. These arcs are shown to sweep whichever integer they contain within their superposition. For example, the superposition obtained from $|6\rangle \mapsto |C_{6,k}\rangle$ (shown as the red arc) sweeps the first $2^k - 1 = 7$ integers whose discrete logarithm come after $log_{11}(6) = 26 \mod p$. Note that the green arc (YES feature state) overlaps much more with the blue arc (normal to the hyperplane) than does the red arc (NO feature state), highlighting the role of $|\phi_s\rangle$ as defining a wide-margin hyperplane in this space.

which as illustrated in figure 1b produces a different value depending on the class label of y:

$$= \begin{cases} 2^k & \text{for } (1-\Delta)\% \text{ of } \Pi_Y^{\text{DLP}} \\ 0 & \text{for } (1-\Delta)\% \text{ of } \Pi_N^{\text{DLP}} \end{cases}$$

where $\Delta = \frac{2^k}{(\frac{(p-1)}{2})} = \frac{2^{k+1}}{p-1}$ is the ratio of the arc length spanned by the feature state to that of $|\phi_s\rangle$. By including the normalisation factor and squaring the entire quantity, we get the Hilbert-Schmidt inner product:

$$|\langle\phi_s|C_{y,k}\rangle|^2 = \begin{cases} \Delta & \text{for } (1-\Delta)\% \text{ of } \Pi_Y^{\text{DLP}} \\ 0 & \text{for } (1-\Delta)\% \text{ of } \Pi_N^{\text{DLP}} \end{cases} \quad (5)$$

The fact that taking the inner product with $|\phi_s\rangle$ is able to separate YES and NO is reminiscent of separating hyperplanes in the context of the support vector machine. In fact,

Remark. The state $|\phi_s\rangle$ is the normal to a hyperplane that separates the YES and NO instances of Π^{DLP} when mapped to the feature space $\mathcal{V} = \text{span}\{|C_{y,k}\rangle \ \forall y\}$.

To demonstrate the sense in which $|\phi_s\rangle\langle\phi_s|$ (i.e. $|\phi_s\rangle$ as a density matrix) is the normal vector to a differentiating hyperplane, we can alternatively represent both it and additionally any $|C_{y,k}\rangle\langle C_{y,k}|$ with respect to the 4^n-dimensional Pauli basis $\{P_\alpha\}$. Using the HS inner product, we identify these density matrices with their Fourier coefficients, $\vec{w}_s = (w_{s,\alpha})$ and $\vec{y} = (y_\alpha)$, respectively. In this representation, the HS inner product of the space $\mathbb{C}^{2^n \times 2^n}$ reduces to the familiar Euclidean inner product in $\mathbb{R}^{4^n}$:

$$\langle|\phi_s\rangle\langle\phi_s|\,, |C_{y,k}\rangle\langle C_{y,k}|\rangle_{HS} = |\langle\phi_s|C_{y,k}\rangle|^2$$
$$= \sum_{\alpha=0}^{4^n-1} w_{s,\alpha} y_\alpha = \langle\vec{w}_s, \vec{y}\rangle_{\mathbb{R}} \,.$$

Taking $b = \frac{\Delta}{2}$, the inner products in 5 can be recast as Euclidean vectors:

- $\vec{w}_s \cdot \vec{y} - b = \frac{\Delta}{2}$ for $(1-\Delta)\%$ of Π_Y^{DLP}

- $\vec{w}_s \cdot \vec{y} - b = -\frac{\Delta}{2}$ for $(1-\Delta)\%$ of Π^{DLP}_{Y}.

where $\vec{w}_s$ and b can naturally be interpreted as the normal vector to a hyperplane and offset term respectively; at least $(1-\Delta)\%$ of the YES (NO) data points lie above (below) the hyperplane. Thus, this hyperplane separates the classes with a (soft) margin of $\Delta/\|w_s\|$.

2.3.1 Implementing the quantum feature map $|y\rangle \mapsto |C_{y,k}\rangle$

As a first step towards building the desired quantum feature map in Equation 3, consider the following classical operation

$$\begin{aligned} C_{y,k} &: \{0,1\}^k \to \{0,1\}^n \\ C_{y,k}(i) &= y \cdot g^i \mod p\,. \end{aligned}$$

By using elementary multiplication modulo p, the quantum circuit for which is well understood and efficient[3], it is easy to implement the reversible operator

$$\begin{aligned} \hat{C}_{y,k} &: \mathcal{H}^{\otimes 2n} \to \mathcal{H}^{\otimes 2n} \\ \hat{C}_{y,k}\,|i\rangle\,|0\rangle^{\otimes n} &= |i\rangle\,|C_{y,k}(i)\rangle \end{aligned}$$

where $|C_{y,k}(i)\rangle$ in the second register acts on the k least significant bits from the first register. Now using the exact quantum algorithm for solving the DLP, the following operator can be reproduced REF:

$$U_y\,|C_{y,k}(i)\rangle\,|0\rangle^{\otimes k} = |i\rangle\,|C_{y,k}(i)\rangle \tag{6}$$

Finally, this operator enables the desired superposition state to be obtained up to the addition and removal of auxiliary qubits:

$$|0\rangle \xrightarrow{H^{\otimes k}} \sum_{i=0}^{2^k-1} |i\rangle \xrightarrow{\hat{C}_{y,k}} \sum_{i=0}^{2^k-1} |i\rangle\,|C_{y,k}(i)\rangle \tag{7}$$

$$\xrightarrow{U_y^{\dagger}} \sum_{i=0}^{2^k-1} |C_{y,k}(i)\rangle := |C_{y,k}\rangle \tag{8}$$

Classical hardness of learning Π^{DLP} To make provable statements, Liu *et al.* use the learning theoretic notion of a concept class as opposed to that of a promise problem when defining their DLP-based dataset. In particular, the DLP concept class is the set of labeling rules $C = \{h_{\mathrm{DLP},s} \forall s \in \mathbb{Z}_p^*\}$. Borrowing ideas from PAC learning, they also define what *efficiently learning* a concept class entails. With these precise definitions, they show that if a classical computer were to efficiently learn C, then so too could it efficiently solve the DLP. On the other hand, they show that the task of efficiently learning C is in BQP and that the `QKE-SVM` is one such QML model that can efficiently learn it.

In our paper, we shift away from the PAC-learning definition of *efficient learning* by a `QKE-SVM` and instead relax this definition. We will assume that the `QKE-SVM` can efficiently learn a promise problem if a) there exists a quantum feature space in which the promise problem becomes linearly separable and b) the map into said space can be efficiently implementable. At the cost of compromising on rigour, this definition will help us better characterise the necessary requirements for a promise problem to become efficiently learnable by a `QKE-SVM`.

3 A framework for identifying classically intractable classification tasks admitting quantum speedup

The success of Π^{DLP} can be captured in terms of three characteristics:

C.1 Π^{DLP} is classically intractable.

C.2 There exists a quantum feature space $\mathcal{V}$ in which the Π^{DLP} becomes linearly separable. We say the Π^{DLP} *admits* a linearly separable representation in quantum feature space.

C.3 The quantum feature map $\varphi_{\mathcal{V}}$ and by extension, the kernel function $K_{\mathcal{V}}$ associated with the quantum feature space $\mathcal{V}$ can be computed efficiently.

C.2 and C.3 together imply that the Π^{DLP} can be efficiently solved using a `QKE-SVM`. Subsequently, C.1 suggests that no classical model exists that could solve the Π^{DLP} better than just randomly guessing, therefore completing the claim for quantum speedup.

As summarised in Figure 2, we will see that each of these three characteristics originates from some sufficient corresponding mathematical property of the Π^{DLP}:

[3] Using the multiplication operator $U_{k,p}\,|l\rangle = |k \cdot l \mod p\rangle$ and exponentiation operator $V_{k,p}\,|l\rangle = |k^l \mod p\rangle$, one can at once see that $|C_{y,k}(i)\rangle = U^i_{y,p} V_{g,p}\,|i\rangle$[?]

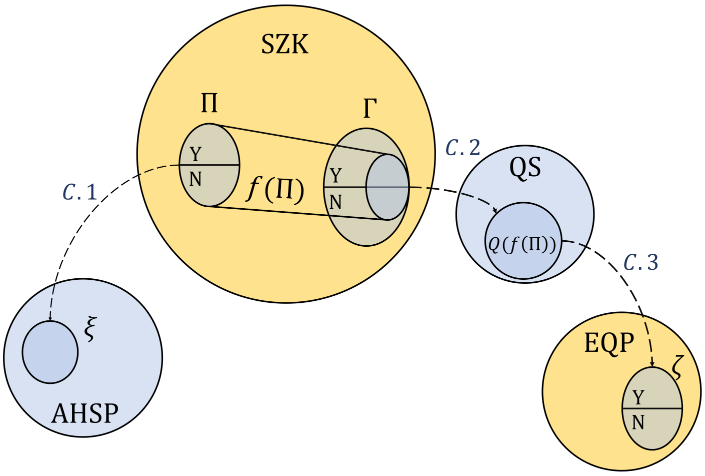


Figure 2: **Diagrammatic representation of a framework for identifying potential classically intractable classification tasks that are learnable by a `QKE-SVM`.** A promise problem Π will admit a quantum feature space in which it becomes linearly separable (**C.2**) if it belongs to SZK. $\Pi \in SZK$ Karp-reduces to the SZK-complete problem $\mathrm{SD}_{\alpha\beta}$, denoted here by Γ [?], which further Cook-reduces to the problem of Quantum Sampling. The QS instance of Π produces our desired quantum feature map. If the QS reduction of Π additionally cook-reduces to a problem ζ within the class of Exact Quantum Polynomial-time algorithms, then we are guaranteed that the quantum feature map above has an efficient implementation on a quantum computer (**C.3**). Finally, if the original promise problem Π also cook-reduces to some Abelian Hidden Subgroup Problem ξ, then it may be *potentially* classically intractable (**C.1**); Π is only strongly classically intractable if it is shown to be equivalent to ξ, which Liu *et al.* prove for the case of their DLP concept class [6].

O.1 Π^{DLP} is equivalent to the DLP which is conjectured to be classically difficult.

O.2 Π^{DLP} belongs to the class of problems that have Statistical Zero Knowledge (SZK) proofs [1].

O.3 The SZK reduction of the Π^{DLP} further reduces to the class of Exact Quantum Polynomial-time (EQP) algorithms REF[?].

Investigating these properties will help to guide the search for other promise problems that could be learned advantageously with a `QKE-SVM` in much the same way as the Π^{DLP}. In what follows, we will derive the properties **O.1**-**O.3** by examining Π^{DLP} as a special case and then discuss them more generally in terms of a framework. This framework will lay the foundations for our subsequent investigation of two new promise problems which extend on the Π^{DLP}.

3.1 Statistical Zero-Knowledge: a class of problems admitting a linearly separable representation in quantum feature space

3.1.1 Differential overlap in Π^{DLP} arises from statistical difference

What mathematical property of the Π^{DLP} allows it to become linearly separable in some quantum feature space? To answer this question, we will have to reinterpret the meaning of the HS inner product $|\langle\phi_s|C_{y,k}\rangle|^2$. Note that the quantity $|\langle\phi_s|C_{y,k}\rangle|$ is the *overlap* between the basis states of $|C_{y,k}\rangle$ and $|\phi_s\rangle$.[4]. We recall from Equation 5 that the inner product $|\langle\phi_s|C_{y,k}\rangle|$ takes on different values based on the class label of y. This qualifies the state $|\phi_s\rangle = |C_{g^s,\lfloor\log_2(p)\rfloor-1}\rangle$ as a hyperplane that separates the YES or NO instances.

We now reinterpret the notion of an *overlap* by returning to the quantum mechanical interpretation of quantum states as probability (amplitude) distributions:

$$|\psi\rangle = \sum_i \lambda_i |i\rangle \implies |i\rangle \text{ measured with probability } |\lambda_i|^2$$

In this perspective, $|C_{y,k}\rangle$ represents a uniform

[4] For a concrete example, observe that $|\psi\rangle = |1\rangle + |3\rangle + |5\rangle$ and $|\phi\rangle = |3\rangle + |4\rangle + |5\rangle$ have an overlap $\langle\psi|\phi\rangle = 2$.

distribution over the values $\{y \cdot g^i \,|\, \forall i \in [0, 2^k - 1]\}$. We may define this probability distribution independently from the quantum state. Indeed, the image of the function

$$C_{y,k} : \{0,1\}^k \to \{0,1\}^n, \quad i \mapsto y \cdot g^i \mod p \tag{9}$$

has the associated probability distribution

$$D_{C_{y,k}} : \{0,1\}^n \to \{0,1\}, \quad x \mapsto \frac{|C_{y,k}^{-1}(\{x\})|}{2^k} \tag{10}$$

In words, $D_{C_{y,k}}(x)$ refers to the proportion of inputs i in the domain of $C_{y,k}$ whose image is $C_{y,k}(i) = x$. Since $C_{y,k}$ is a bijection, there is a unique input corresponding to each output which implies $D_{C_{y,k}}(x) = \frac{1}{2^k}$ over its support. To summarise, we have recast the feature state $|C_{y,k}\rangle$ as the probability distribution $D_{C_{y,k}}$, with both quantities being related to each other in the following way:

$$|C_{y,k}\rangle = \sum_{x \in \{0,1\}^n} \sqrt{D_{C_{y,k}}(x)}\, |x\rangle \tag{11}$$

If instead of quantum states $|C_{y,k}\rangle$ and $|\phi_s\rangle$ we worked with their probability distribution representations, $D_{C_{y,k}}$ and $D_{C_{g^s, \lfloor \log_2(p) \rfloor - 1}}$, then the analogue of the differential overlap we saw in (5) is the difference between the two probability distributions:

$$|D_{C_{y,k}} - D_{C_{g^s, \lfloor \log_2(p) \rfloor - 1}}| \begin{cases} \geq \alpha & \text{for } (1-\Delta)\% \text{ of } \Pi_Y^{\text{DLP}} \\ \leq \beta & \text{for } (1-\Delta)\% \text{ of } \Pi_N^{\text{DLP}} \end{cases} \tag{12}$$

for some $0 < \beta < \alpha^2 < 1$. In [1], it is shown that the statistical difference and HS inner product tightly bound each other. This representation of statistical difference inspires a new promise problem which the Π^{DLP} reduces[5] to:

$$\begin{aligned} \Gamma^{\text{DLP}} &= (\Gamma_Y^{\text{DLP}}, \Gamma_N^{\text{DLP}}) \qquad (13) \\ \Gamma_Y^{\text{DLP}} &= \left\{ (C_{y,k}, C_s) \ : \ |D_{C_{y,k}} - D_{C_s}| \geq \alpha \right\} \\ \Gamma_N^{\text{DLP}} &= \left\{ (C_{y,k}, C_s) : \ |D_{C_{y,k}} - D_{C_s}| \leq \beta \right\}, \end{aligned}$$

where C_s is shorthand for $C_{g^s, \lfloor \log_2(p) \rfloor - 1}$. Even though we have derived (??) from the differential quantum overlap in (5), we could alternatively interpret the relationship in reverse: Γ^{DLP} suggests a differential quantum overlap. This implies that other problems that can be cast as a statistical difference problem admit a linearly separable representation in a quantum feature space. Conveniently, there is a class of problems that all reduce to this instance.

3.1.2 All Statistical Zero-Knowledge (SZK) problems reduce to Statistical Difference $\text{SD}_{\alpha\beta}$

The promise problem in **??** is an instance of a general problem known as Statistical Difference $\text{SD}_{\alpha\beta}$:

$$\begin{aligned} \Gamma &= (\Gamma_{\text{Y}}, \Gamma_{\text{N}}) \\ \Gamma_{\text{Y}} &= \{(f,g) \ : \ |D_f - D_g| > \alpha\} \\ \Gamma_{\text{N}} &= \{(f,g) \ : \ |D_f - D_g| < \beta\} \end{aligned}$$

where $0 \leq \beta \leq \alpha \leq 1$ satisfying $\alpha^2 \geq \beta$.

The upshot of this formulation is that there is an entire class of problems that reduces to this very problem. This is the class of problems admitting a *statistical zero-knowledge (SZK) proof.*

Remark. $\text{SD}_{\alpha\beta}$ is SZK-complete

Informally, a SZK problem consists of an interactive proof system whereby a *verifier* tries to verify that an element x satisfies some condition P via a two-way exchange of messages between itself and an all-knowing *prover*. For a problem to be categorised as SZK, however, the verifier must not learn anything during its exchange with the prover other than the fact that x satisfies P [?]. In fact, Π^{DLP} reduces to Γ^{DLP} precisely because Π^{DLP} is in SZK. Besides this problem, the SZK contains other problems including Quadratic Residuoisity (QR) and the Closest Vector Problem (CVP) often used in cryptographic contexts.

We have shown that the SZK problem Π^{DLP} has differential quantum overlap because of its reduction to $\text{SD}_{\alpha\beta}$. Can we show the same for other SZK problems?

As we saw in 11 for the case of $D_{C_{y,k}}$ and $|C_{y,k}\rangle$, to go from statistical difference to a quantum differential overlap, the functions $(f,g) \in \Gamma^{\text{DLP}}$ must be efficiently prepared as quantum states

$$|f\rangle = \sum_{z \in \text{Im}(f)} \sqrt{D_f(z)}\, |z\rangle \tag{14}$$

[5] But it doesn't? C_s can't be efficiently generated without knowledge of s which is what is being learned. Unless we set $s = \frac{p}{2}$ in which case the problem is still hard

This is referred to as the general task of Quantum Sampling (QS) in which a classical circuit f is given and the goal is to produce a corresponding quantum circuit which encodes the distribution D_f as a superposition of states.

The reduction of SZK to QS in fact *induces* a quantum feature map into some quantum feature space wherein the SZK problem is linearly separable. In concrete terms, if $x \in \Pi$ is a specific instance of some SZK language Π, the procedure $x \mapsto |f_x\rangle$ can be interpreted as a quantum feature map in analogy to $y \mapsto |C_{y,k}\rangle$. This satisfies C.2. In the next section we will discuss sufficient criteria that allow efficient implementation of these feature maps.

3.2 EQP for efficient index erasure and the preparation of quantum feature maps

In the previous section we established that for every promise problem belonging to the class of Statistical Zero Knowledge problems, there exists a quantum feature space in which the promise problem becomes linearly separable. However, it is a different question entirely to ask whether there exists an *efficient* mapping into that induced quantum feature space. To investigate this, let us once again begin with Liu *et al.*'s special case: what mathematical property of the Π^{DLP} allows the quantum feature *map* in Eq. (3) to be efficiently evaluated using a quantum computer?

The series of operations from Eq. (8) used to obtain the feature state $|C_{y,k}\rangle$ can be divided into two main components:

$$\underbrace{|0\rangle \xrightarrow{H^{\otimes k}} \sum_{i=0}^{2^k-1} |i\rangle \xrightarrow{U_{\text{mod}}} \sum_{i=0}^{2^k-1} |i\rangle\,|C_{y,k}(i)\rangle}_{\text{Initial state preparation}}$$
$$\underbrace{\xrightarrow{U_y^\dagger} \sum_{i=0}^{2^k-1} |C_{y,k}(i)\rangle}_{\text{Index erasure}} := |C_{y,k}\rangle \ .$$

To elucidate what the steps of initial state preparation and index erasure each entail, we will consider 8 in a slightly more general form. Let us replace the function $C_{y,k} : \{0,1\}^k \to \{0,1\}^n$ as described in (10) with an arbitrary function $f : X \to Y$. In this case, (8) is generalised to

$$\underbrace{|0\rangle \xrightarrow{H^{\otimes k}} \sum_{i=0}^{2^k-1} |i\rangle \xrightarrow{U_{\text{sift}}} \sum_{i\in X} |i\rangle \xrightarrow{U_f} \sum_{i\in X} |i\rangle\,|f(i)\rangle}_{\text{Initial state preparation}}$$
$$\underbrace{\xrightarrow{U_{\text{uncomp}}} \sum_{i\in X} |f(i)\rangle}_{\text{Index erasure}} := |f\rangle \ . \tag{15}$$

It is clear that the initial state preparation step is governed by the subsequent step of index erasure. For that reason, let us examine index erasure with respect to the arbitrary function f more closely, referring back to the special case of the Π^{DLP} where $f = C_{y,k}$ only where it consolidates understanding.

Index erasure refers to the task of uncomputing ('erasing') the input ('index') of f from a superposition of paired input-output states *i.e.* (**??**). Whether there exists an efficient unitary U_{uncomp} capable of erasing the index of the state (**??**) is in general a nontrivial problem and completely dependent on the function f. Constructing U_{uncomp} typically involves finding its adjoint

$$|f(i)\rangle \xrightarrow{U^\dagger_{\text{uncomp}}} \sum_{j\in f^{-1}(\{f(i)\})} |j\rangle\,|f(i)\rangle \tag{16}$$

which in the special case of the Π^{DLP} corresponds to

$$|C_{y,k}(i)\rangle \xrightarrow{U_y} |i\rangle\,|C_{y,k}(i)\rangle \ ,$$
$$|g^{\log_g(y)+i}\rangle \xrightarrow{U_y} |i\rangle\,|g^{\log_g(y)+i}\rangle \ ,$$

such that $U^\dagger_{\text{uncomp}} := U_y$. Observe that U_y computes the discrete logarithm[6] which quantum computers are known to perform efficiently. The DLP is a member of the Abelian hidden subgroup problems (AHSP), all of which are efficiently solvable in BQP time using a quantum computer [?]. Note, however, that U_y calculates the discrete logarithm *exactly* and not with bounded-error. This is because the DLP additionally belongs to the class of Exact Quantum Polynomial (EQP). The process of developing an exact quantum algorithm from the bounded-error version is refered to as derandomisation. More recently, other problems in AHSP have been derandomised and we will use these in subsequent sections.

[6] up to division by y

Abelian hidden subgroup problems usually involve some *one-way* function f that is easy to compute on every input but whose preimage is computationally hard to calculate. In the case of the Π^{DLP}, observe that $C_{y,k}$ is efficient to compute on inputs i, but calculating the preimage $C_{y,k}^{-1}(\{x\}) = C_{y,k}^{-1}(x) = \log_g(x)$ is conjectured to be classically difficult.

What about the general case of any arbitrary one-way function f that does not originate from the AHSP? the most naive approach to index erasure in this case would involve explicitly computing $f(i) \forall i \in X$ and building a lookup table, which would take $\mathcal{O}(|X|)$ time and space. Instead, in Algorithm 2, we provide an algorithm for index erasure based on amplitude amplification that takes $\mathcal{O}(\sqrt{|X|})$ number of steps, affording a quadratic speedup from the naive approach. Of course, the ideal case is if f belongs to the AHSP where index erasure can be computed exponentially faster, $\mathcal{O}(\log(|X|))$.

In summary, the quantum feature map for a SZK promise problem can be efficiently implemented if it admits efficient index erasure as in the Π^{DLP}. This occurs if the QS reduction of the SZK promise problem – which may be harder than the promise problem itself – also belongs to AHSP $\cap$ EQP. This satisfies criteria **C.3**.

It is important not to confuse **C.3** and **C.1**. In **C.1**, the fact that the original SZK promise problem Π^{DLP} reduces to AHSP renders Π^{DLP} classically intractable. In **C.3**, the fact that the *QS reduction* of Π^{DLP} reduces to AHSP $\cap$ EQP (which need not follow from **C.1**) ensures the efficient implementation of the quantum feature map on a quantum computer.

In the next section, we will identify and investigate two new SZK promise problems. The first promise problem belongs to the AHSP but has no efficient initial state preparation, while the second promise problem has an efficient initial state preparation, but does not belong to the AHSP.

3.3 Summary and reflections

By closely examining the Π^{DLP}, we have identified that promise problems belonging to a class known as SZK admit quantum feature spaces in which they become linearly separable **C.2**. This is because all SZK problems reduce to a complete problem known as STATISTICAL DIFFERENCE $\text{SD}_{\alpha\beta}$ and consequently QUANTUM SAMPLING, which relates distances between probability distributions to differential inner products between quantum states. If the QUANTUM SAMPLING reduction of the SZK problem itself reduces to EQP, we are guaranteed an efficient quantum map into that feature space, whether it uses index erasure or not. Together, these two features allow the promise problem to be learned by a `QKE-SVM`. Finally, if the SZK promise additionally belongs to the class of AHSP then it can be strongly conjectured that the problem is classically intractable, enabling the claim to be made that the `QKE-SVM` is demonstrating quantum advantage.

4 A classification task based on quadratic residuosity with unknown index erasure

Guided by the framework we proposed in the previous section, we will now investigate another famous problem in SZK called the Quadratic Residuosity problem as a potential classification task to be learned by the `QKE-SVM` [1]. We will introduce the notion of quadratic residues from number theory first and then design a promise problem Π^{QRP} that is based directly on it. While there exists a quantum feature map in which Π^{QRP} is linearly separable, we will see that no efficient implementation into this space is known because the promise problem is not known to satisfy **O.3**.

4.1 Quadratic residues

Let $\mathbb{Z}_N^*$ be the multiplicative group of integers modulo $N = p \times q$, where p and q are large primes. The number of elements in $\mathbb{Z}_N^*$ is given by Euler's totient function $\varphi(N) = (p-1)(q-1)$ which counts the number of integers less than N that are relatively prime to N[7].

In number theory, an element $x \in \mathbb{Z}_N^*$ is called a *quadratic residue* or perfect square if there exists a $y \in \mathbb{Z}_N^*$ such that $x = y^2 \mod N$. We say that xRN if x is a quadratic residue. The task of identifying whether a given x is a quadratic residue is classically difficult and to date no quantum operation is known that can solve it exponentially faster [**?**]. From the Chinese remainder

[7] for example, $N = 15$ is coprime with $1, 2, 4, 7, 8, 11, 13, 14$, so $\varphi(15) = (3-1)(5-1) = 8$

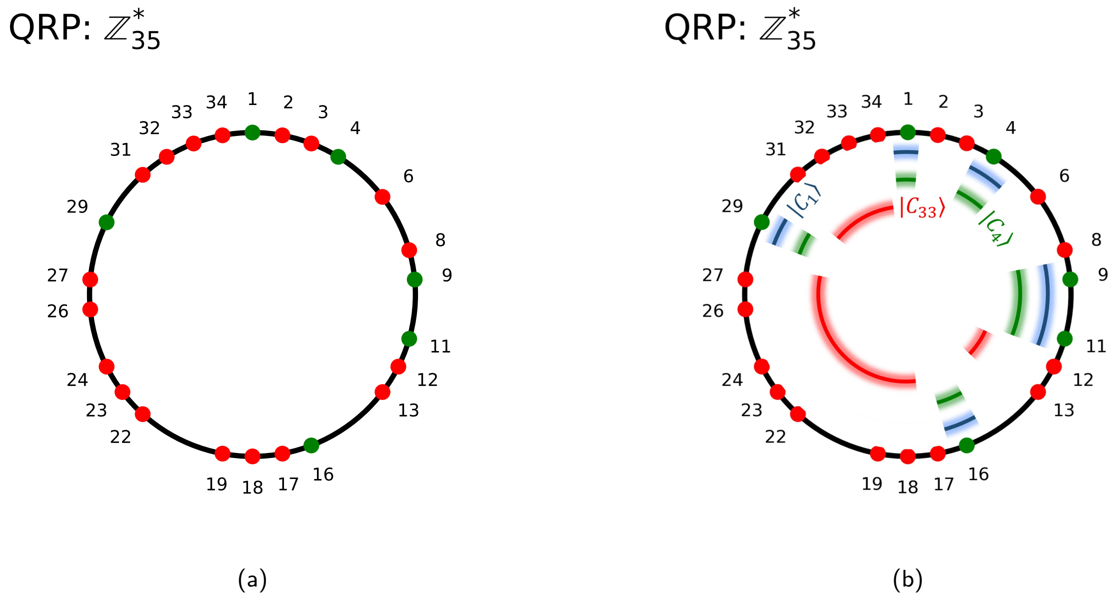


Figure 3: **Use of the quantum feature map 18 to separate the yes/no (green/red) instances of Π^{OFP} for the special case of $N = 35$.** In dataset 3b, the superposition (shown in blue) obtained as $1 \mapsto |C_1\rangle$ sweeps all quadratic residues (*c.f.* Eq (18)), which corresponds to all the green dots $\in \Pi^{\mathrm{QRP}}_Y$. Analogous to 1b, the blue, green and red arcs indicate the normal to the hyperplane, the feature state of $4 \in \Pi^{\mathrm{QRP}}_Y$, and the feature state of $33 \in \Pi^{\mathrm{QRP}}_N$ respectively. Note that in all cases, the green arcs (feature states) overlap much more with the hyperplane than do the red arcs, indicating that the separating hyperplane has a wide margin.

theorem, it is known a priori that of the $\phi(N)$ elements of $\mathbb{Z}^*_N$, the proportion of quadratic residues is exactly a 0.25. As a result, a benchmark ZeroR classifier can at best achieve a 75% accuracy on this dataset.

Naturally, the promise problem Π^{QRP} consists of the language of integers that are quadratic residues of N.

$$\begin{aligned}\Pi^{\mathrm{QRP}} &= (\Pi^{\mathrm{QRP}}_Y, \Pi^{\mathrm{QRP}}_N) \\ \Pi^{\mathrm{QRP}}_Y &= \{x : \; xRN\} \\ \Pi^{\mathrm{QRP}}_N &= \{x : \; \neg xRN\} \;,\end{aligned} \tag{17}$$

Figure 3a displays a fully labelled set of $\mathbb{Z}^*_N$ for $N = 35 = 5 \cdot 7$ according to the rule above. Note that the occasional gap indicates an integer that is not coprime with N and is thus not a member of the set. Also notice that exactly 25% of the data points are labelled as quadratic residues, reflecting the proportion of quadratic residues in $\mathbb{Z}^*_N$ derived earlier.

4.2 A quantum feature map for separating Π^{QRP}.

It is well known that Π^{QRP} belongs to SZK. When reduced from SZK to QUANTUM SAMPLING, Aharonov was able to show that Π^{QRP} is linearly separable in the following quantum feature space [1]:

$$x \mapsto \sum_{r \in \mathbb{Z}^*_N} |r^2 x \mod N\rangle := |C_x\rangle \;. \tag{18}$$

Compared to the feature map in Π^{DLP}, it seems less clear why this quantum feature map (18) admits a separating hyperplane. Nevertheless, we will show that $|C_1\rangle$ is a hyperplane that separates the YES and NO instances from each other with a large margin.

We now make use of two theorems which are proved in [1]:

Theorem 1. *$x \in \Pi^{QRP}_Y \iff \forall r \in \mathbb{Z}^*_N$, $C_x(r) = r^2 x \in \Pi^{QRP}_Y$.*

Theorem 2. *$x \in \Pi^{QRP}_Y \implies \forall z \in \Pi^{QRP}_Y \exists r \in \mathbb{Z}^*_N$ such that $z = r^2 x \mod p$. That is, C_x maps surjectively to the set of all quadratic residues* mod N.

From Theorem 1, it follows that if x is a quadratic residue, then so are all the basis states of $|C_x\rangle$. Additionally, from Theorem 2 we know that for every quadratic residue $z \in \Pi^{\mathrm{QRP}}_Y$, there will be a corresponding basis state $|C_x(r)\rangle = |z\rangle$. Therefore, $|C_x\rangle$ is a uniform superposition of all quadratic residues in $\mathbb{Z}_N^*$, $\sum_{x\in\Pi_Y^{\mathrm{QRP}}}|x\rangle$. But this is exactly the definition of $|C_1\rangle$:

$$|C_1\rangle = \sum_{r\in\mathbb{Z}_N^*}|r^2\rangle = \sum_{z\in\Pi_Y^{\mathrm{QRP}}}|z\rangle = |C_x\rangle \tag{19}$$

So, $\langle C_1|C_x\rangle = 1$ and the states completely overlap.

Alternatively, if x is a quadratic nonresidue, then by Theorem 1, neither is $r^2x\,\forall r$, so that $|C_1\rangle$ and $|C_x\rangle$ share no overlap and $\langle C_1|C_x\rangle = 0$. To summarise:

$$|\langle C_1|C_x\rangle|^2 = \begin{cases} = 1 & \forall x \in \Pi_Y^{\mathrm{QRP}} \\ = 0 & \forall x \in \Pi_N^{\mathrm{QRP}}. \end{cases} \tag{20}$$

Preparing the quantum feature map for the QR_P. As in the case of Π^{DLP}, one approach to preparing the feature map $|C_x\rangle$ in Equation (18) relies on index erasure:

$$\sum_{i\in\mathbb{Z}_N^*}|i\rangle\,|i^2 \mod N\rangle \xrightarrow{U_{\mathrm{uncomp}}} \sum_{i\in\mathbb{Z}_N^*}|i^2 \mod N\rangle$$
$$\xrightarrow{U_{\times}} \sum_{i\in X}|i^2\cdot x \mod N\rangle = |C_x\rangle$$

While the modular multiplication step, $U_{\times}$, has a well-known efficient implementation, it is not clear how one would obtain U_{uncomp}. If we had an efficient modular root operator,

$$|0\rangle\,|i^2 \mod N\rangle \xrightarrow{U_{\surd}} (|i\rangle+|N-i\rangle)\,|i^2 \mod N\rangle\,, \tag{21}$$

then we could easily recover $U_{uncomp} = U_{\surd}^{\dagger}$. Indeed, an efficient operator $U_{\surd}$ would be able to solve the quadratic residue problem, the conjectured difficulty of which has served it well as a robust encryption protocol. Indeed, the QRP does not belong to the AHSP either, so finding an efficient quantum feature map for the state above on that basis is difficult. If we are willing to surrender exponential speedup, we may still obtain a quadratic improvement by using the general-purpose index erasure algorithm in algorithm 2. We propose algorithm 3 to compute U_{uncomp} with quadratic speedup $O(\sqrt{|\mathbb{Z}_n^*|})$ [? ?].

The Π^{QRP} admits a quantum feature space wherein it become linearly separable, but has no known efficient quantum feature map into it. The Π^{QRP} has no efficient algorithm for index erasure, which highlights the importance of the QS instance belonging to EQP for it to be efficiently implementable as a unitary using a quantum computer.

5 A new classically intractable classification task based on order finding that circumvents index erasure

In the previous section, we encountered a promise problem in SZK whose corresponding quantum feature map lacked a known efficient implementation that does not depend on index erasure. This raises a question: are there promise problems that are classically intractable, efficiently learnable by the `QKE-SVM`, but importantly do not require index erasure?

We answer this question in the affirmative by presenting a novel promise problem Π^{OFP} that crucially avoids index erasure and still satisfies criteria **O.1–O.3**. We show that the Π^{OFP} is at most as difficult as order finding, and argue as to why we may conjecture that it is classicaly intractable. Our main result will be proving that Π^{OFP} belongs to SZK and as a consequence, admits a quantum feature space in which it becomes linearly separable. Unlike Π^{QRP} from the previous section, we will discover that the quantum feature map for Π^{OFP} will be efficient to implement chiefly because it circumvents index erasure. Our promise problem is able to avoid index erasure because it belongs to a version of SZK where the verifier is allowed access to a quantum computer. Thus, Π^{OFP} remains classically intractable all while admitting an efficiently implementable quantum feature map that can linearly separate it. By way of simulations, we demonstrate that the `QKE-SVM` outperforms classical models in learning this promise problem because it has access to the corresponding quantum feature space.

In what follows, we introduce some elementary results from number theory that are necessary to define Π^{OFP}. Here, we also prove a new result about the distribution of odd and even orders of large semiprimes. After laying this foundation, we present the promise problem Π^{OFP} and show that it satisfies criteria **C.1** - **C.3** which constituted our framework from section **??**. We present a quantum feature map which separates the YES and NO instances of Π^{OFP} with a large margin and furthermore boasts an efficient implementation. To support our technical findings, we conduct simulations illustrating that various classical models struggle to learn the dataset induced by Π^{OFP} as effectively as the `QKE-SVM`.

5.1 Preliminaries

The Π^{OFP} promise problem will be largely based on the properties of multiplicative orders which we will now briefly define in this section. In the following, $N = pq$ is a product of two primes, also known as a semiprime.

Definition 5.1 (multiplicative order modulo N)**.** The multiplicative order (or simply the order) of an element $a \in \mathbb{Z}_N^*$ is defined as the smallest positive integer r satisfying $a^r = 1 \mod N$. We denote the multiplicative order by $r = ord_n(a)$.

We also state a well-known property of orders without proof

Theorem 3. *For all $a \in \mathbb{Z}_N^*$, the order $ord_N(a)$ divides the totient $\varphi(N) = (p-1)(q-1)$.*

Our promise problem will also make use of a property we have called the evenness:

Definition 5.2 (Evenness $\mathcal{E}$)**.** The *evenness* of an integer w, denoted by $\mathcal{E}(w)$, is the power of 2 that occurs in the factorisation of w. If $w = 2^a m$ is the partial factorisation of w, where $m \geq 1$ is the largest odd factor and $a \geq 0$, then by definition $\mathcal{E}(k) = a$.

We now state the following theorem which will serve as the main foundation for our promise problem. The proof of this theorem can be found in section B of the Appendix.

Theorem 4. *If $N = pq$ is a semiprime with $\mathcal{E}(\varphi(N)) = a$, then the number of elements in $\mathbb{Z}_N^*$ whose multiplicative orders are odd is $\frac{\varphi(N)}{2^a}$.*

Corollary 4.1. *Fermat semiprimes which are defined as $S = p'q'$ where $p' = 2^{2^{k_1}} + 1$ and $q's = 2^{2^{k_2}} + 1$ are Fermat primes have no elements with odd orders except the trivial element 1.*

Proof. For a fermat prime

$$\varphi(S) = (2^{2^{k_1}})(2^{2^{k_2}}) = 2^{2^{k_1}+2^{k_2}} = 2^{a'} \tag{22}$$

Thus, by Theorem 4, the number of elements with odd orders is $\varphi(S)\left(\frac{1}{2^{a'}}\right) = 1$ which can only be the trivial element 1. □

5.2 The Π^{OFP} promise problem

The instances in Π^{OFP} are large odd semiprimes $N = pq \in \mathbb{Z}$ which are labelled either YES or NO according to the labeling rule 0

$$h(N) = \begin{cases} +1, & \mathcal{E}(\phi(N)) = 2 \\ -1, & \mathcal{E}(\phi(N)) \geq 9 \end{cases} \tag{23}$$

Giving rise to the promise problem

$$\begin{aligned} \Pi^{\text{OFP}} &= (\Pi_Y^{\text{OFP}}, \Pi_N^{\text{OFP}}) \\ \Pi_Y^{\text{OFP}} &= \{N \ : \ h(N) = +1\} \\ \Pi_N^{\text{OFP}} &= \{N \ : \ h(N) = -1\} . \end{aligned} \tag{24}$$

The labeling rule h assigns to a given odd semiprime N a label based on the evenness of its totient $\varphi(N)$. Specifically, if $\varphi(N)$ is promised to have an evenness of exactly 2, we assign the label +1. Alternatively, if $\varphi(N)$ has an evenness of ≥ 9, we assign the label -1. It is worth noting that 2 represents the minimum evenness that $\varphi(N)$ can have for a semiprime N.

5.3 Π^{OFP} is a potentially classically intractable promise problem learnable by a `QKE-SVM`

In this section, we examine Π^{OFP} according to criteria **O.1-O.3** which comprise our framework for building difficult promise problems learnable by a `QKE-SVM`

5.3.1 Π^{OFP} is potentially classically intractable

Π^{OFP} can be sufficiently solved if we can efficiently compute the evenness of the totient $\varphi(N)$. If we have prior knowledge of $\varphi(N)$, it is possible to efficiently determine the evenness of $\varphi(N)$ in $\mathcal{O}(\log N)$ by repeatedly halving $\varphi(N)$ until we obtain a non-integer result. The calculation of $\varphi(N)$ itself is known to be equivalent to the factorization of N. Therefore, the task of computing the evenness of $\varphi(N)$ can be reduced to the factorization of N. This reduction implies that determining the evenness of $\varphi(N)$ is, at most, as difficult as factorizing integers, which is a challenging problem in classical computation and furthermore belongs to AHSP, as required by **O.1** of our framework.

If determining the evenness of $\varphi(N)$ is indeed as difficult as factorization, then it is strongly conjectured to be classically intractable. However, it is likely easier since calculating $\mathcal{E}(\varphi(N))$

only provides information about the exponent of 2 in the prime factorization of $\varphi(N)$, and not the exponents of other prime factors which would be required to fully reconstruct $\varphi(N)$. From theorem 4, we know that $\mathcal{E}(\varphi(N))$ directly determines the relative proportion of elements with odd or even orders. If there was a way to randomly sample multiplicative orders modulo N, then we could with high probability decide Π^{OFP}. Still, this would not be an exact method.

Ultimately, whether the evenness of $\varphi(N)$ can be accurately calculated using classical methods without explicitly computing $\varphi(N)$ itself remains an unsolved problem in the literature. Based on this, we conjecture that determining the evenness of $\varphi(N)$ is classically intractable. By the same token, we make the assumption that Π^{OFP}, which involves the even easier task of determining whether the evenness of $\varphi(N)$ is greater than or equal to 9 or equal to 2 is also classically intractable.

5.3.2 There is a quantum feature map that separates Π^{OFP}

Π^{OFP} belongs to SZK. The formulation of Π^{OFP} might at first glance seem contrived: what is the significance of the evenness of the totient of a semiprime N? However, a look at theorem 4 quickly addresses this ambiguity: $\mathcal{E}(\varphi(N))$ directly determines the proportions of elements with even or odd orders. In particular, the higher the evenness[8], the higher the proportion of elements with even orders. This relationship is summarised in the following functino which is parameterised by N:

$$\bar{C}_N : [0, N] \to [0, N] \times \{-1, 0, +1\},$$
$$i \mapsto (i, P(ord_N(i)))$$

$$\bar{C}_N(i) = \begin{cases} (i, \ \ 0) & \text{for } G := N - \varphi(N) \quad \text{elements} \\ (i, +1) & \text{for } B := \varphi(N)(1 - \frac{1}{2^a}) \quad \text{elements} \\ (i, -1) & \text{for } R := \varphi(N)(\frac{1}{2^a}) \quad \text{elements} \end{cases} \tag{25}$$

The proportion of blue (even), red (odd) and green (noncoprime) elements has been depicted in

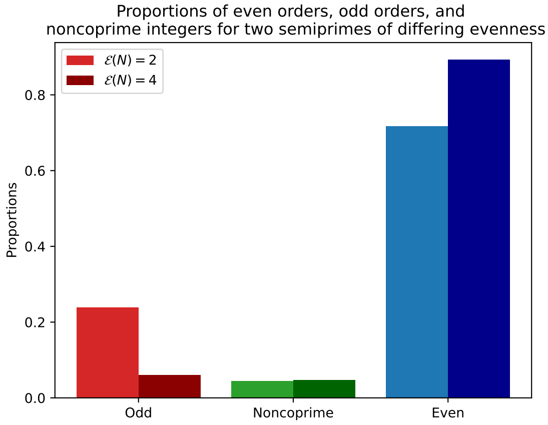


Figure 4: **The distribution of elements which are noncoprime, or otherwise have either odd or even multiplicative orders modulo two semiprimes of evenness 2 and 4.** As proven in Theorem 4, the proportion of even orders is greater for the semiprime with the larger evenness. Note that the semiprimes were chosen to have similar proportion of noncoprimes to highlight theorem 4.

the bar graph in Figure 4 for two semiprimes with differing evenness. Observe that a semiprime of evenness 4 clearly has a greater proportion of even orders (in blue) than a semiprime with evenness 2.

By extension, the YES and NO instances of Π^{OFP} have an even greater difference in the distribution of their order parities. This property should sound familiar. We recall that all SZK problems reduce to STATISTICAL DIFFERENCE, in which a YES (NO) instance is a pair of functions whose induced distributions (as defined in **??**) are largely different (similar). Drawing from Figure 4, we might be tempted to claim that $\bar{C}_S$ and $\bar{C}_N$ form one such pair of functions whose induced distributions are largely dissimilar when N is YES and vice-a-versa (given S itself has a high evenness and is, say, a NO instance):

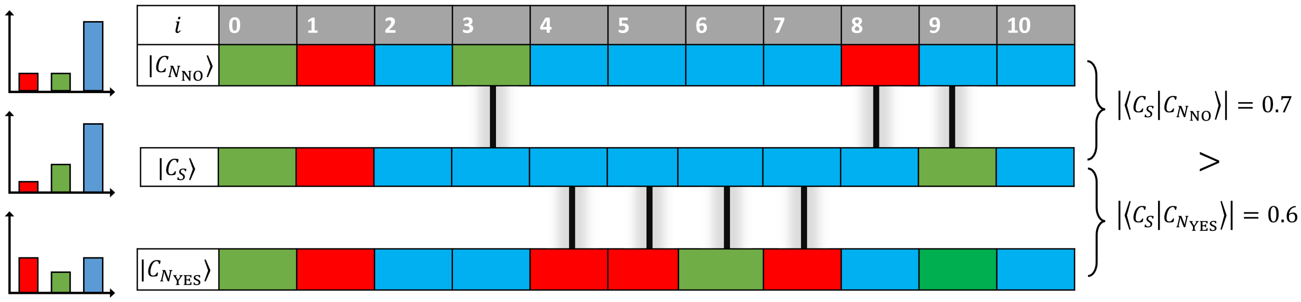
$$\begin{aligned} |D_{\bar{C}_N} - D_{\bar{C}_S}| &\overset{?}{>} \alpha \quad \forall N \in \Pi^{\mathrm{OFP}}_Y \\ |D_{\bar{C}_N} - D_{\bar{C}_S}| &\overset{?}{<} \beta \quad \forall N \in \Pi^{\mathrm{OFP}}_N \end{aligned} \tag{26}$$

for some $0 \leq \beta < \alpha^2 \leq 1$.

However, we must take precautions as the induced distribution $D_{\bar{C}_N}$ has values that depend on N and not solely the evenness. For example, note that

$$D_{\bar{C}_N}((i,j)) = \frac{1}{N} \ or \ 0 \quad \forall i \tag{27}$$

[8] unless stated otherwise, by evenness we mean the evenness *of the totient of a semiprime*, which we omit from here onwards for brevity



i
0
1
2
3
4
5
6
7
8
9
10
$|C_{N_{\text{NO}}}\rangle$
$|C_S\rangle$
$|C_{N_{\text{YES}}}\rangle$
$|\langle C_S|C_{N_{\text{NO}}}\rangle| = 0.7$
>
$|\langle C_S|C_{N_{\text{YES}}}\rangle| = 0.6$

Figure 5: $|C_S\rangle$ **for any** $S \in \Pi_N^{\textsf{OFP}}$ **is the normal to a hyperplane that separates the instances of** $\Pi^{\textsf{OFP}}$. Each cell corresponds to the value of $C_{(\cdot)}(i)$ as represented in terms of RBG colours according to the same colour scheme used in Figure 4. Because N_{NO} and S have a higher evenness than N_{YES}, they have a higher proportion of blue values (even orders) as indicated by the histograms on the left. A given row may be interpreted as the superposition $|C\rangle$ produced by the feature map 29. In this interpretation, the inner product between any two feature states (rows) corresponds to the overlap of the colours. Here, the black lines represent colour mismatches between such pairs. Since $\langle C_{N_{\text{YES}}}|C_S\rangle$ has a greater proportion of mismatches than $\langle C_{N_{\text{NO}}}|C_S\rangle$, it has less overlap and thus a smaller inner product. In section C we show that the statistical difference in 30 are actually directly related to the inner products $\langle C_N|C_S\rangle$.

This dependency makes the relations 26 also depend on the relative sizes of N and S. One could imagine that for $S \gg N$, the differences become almost indiscernible between YES and NO instances because the support of $D_{\bar{C}_N}$ is much less than the support of $D_{\bar{C}_S}$. We fix this problem by making $\bar{C}_N$ periodic and standardising the size of its domain to a fixed multiple of the largest semiprime $M = kN_{\max} > N$ to to be learned by our `QKE-SVM`:

$$\begin{aligned} C_N : [0, M] &\to [0, M] \times \{-1, 0, +1\}, \\ i \mapsto &(i, P(ord_N(i \mod N))) \end{aligned}$$

This standardisation procedure comes at a cost. If N does not divide $M = kN_{\max}$ (which it almost never does unless $N_{\max} = N$), the proportions in Equation 51 of blue, red and green marbles will be different from what they should be in the aperiodic case $\bar{C}_N$. Nevertheless, the proportions of the periodic case can be brought arbitrarily close to the aperiodic case for sufficiently large k.

Now using C_N in place of $\bar{C}_N$, it can indeed be shown that for sufficiently large N.

$$\begin{aligned} |D_{C_N} - D_{C_S}| &\overset{!}{>} \alpha \quad \forall N \in \Pi_Y^{\text{OFP}} \qquad (28) \\ |D_{C_N} - D_{C_S}| &\overset{!}{<} \beta \quad \forall N \in \Pi_N^{\text{OFP}} \end{aligned}$$

for some $0 \leq \beta < \alpha^2 \leq 1$. Thus,

Theorem 5. *The order-finding promise problem Π^{OFP} outlined in Equation 47 reduces to $SD_{\alpha\beta}$ and thus belongs to* SZK *for all semiprimes N with $\frac{\varphi(N)}{N} > 99.6\%$.*

We refer the reader to section C of the Appendix for the proof of the above theorem, among other relevant details and statistical insights into Π^{OFP}.

The quantum feature map separating $\Pi^{\textsf{OFP}}$ Given that Π^{OFP} belongs to SZK and satisfies **O.2**, by our framework it follows that there exists a quantum feature space in which the instances become linearly separable (c.f. **C.2**). To identify this map, it is sufficient to derive the QUANTUM SAMPLING reduction of Π^{OFP}. Drawing on the $\text{SD}_{\alpha\beta}$ reduction of Π^{OFP} from Equation 28, the QUANTUM SAMPLING reductions asks for C_N as a quantum state:

$$\begin{aligned} N \mapsto &\sum_{i \in [0,M]} |i\rangle\, |P(ord_N(i \mod N))\rangle \qquad (29) \\ &:= |C_N\rangle\ , \end{aligned}$$

From our discussion in the previous section, we saw that C_S where $h(S) = -1$ separated the YES and NO instances via differential statistical difference. Naturally, this gives rise to $|C_S\rangle$ as the canonical normal to the hyperplane. This is visualised for a small example in Figure 5. In section C, it is shown that the statistical differences in 28 are actually equal to the inner products between the hyperplane and the feature states. It follows that $|C_S\rangle$ has a hard margin of $\alpha^2 - \beta^2$:

$$|\langle C_N|C_S\rangle|^2 = |D_{C_N} - D_{C_S}|^2 \begin{cases} \geq \alpha^2 & \forall x \in \Pi_Y^{\text{OFP}} \\ \leq \beta^2 & \forall x \in \Pi_N^{\text{OFP}} \end{cases}. \qquad (30)$$

In the appendix, we show how this margin can be widened further using a phase encoding instead of the basis encoding presented in 29.

5.3.3 The quantum feature map separating Π^{OFP} is efficiently implementable

The hallmark characteristic of the feature map in Equation 29 compared to previous ones we have seen in this paper is its ability to retain the index. Since there exists an exact algorithm for order-finding, preparation of the state $|C_N\rangle$ falls comfortably in EQP. This ensures that Π^{OFP} satisfies **O.3** and affords us efficient feature map implementation which we have described in Algorithm 1.

5.4 Results and simulations

5.5 Discussion

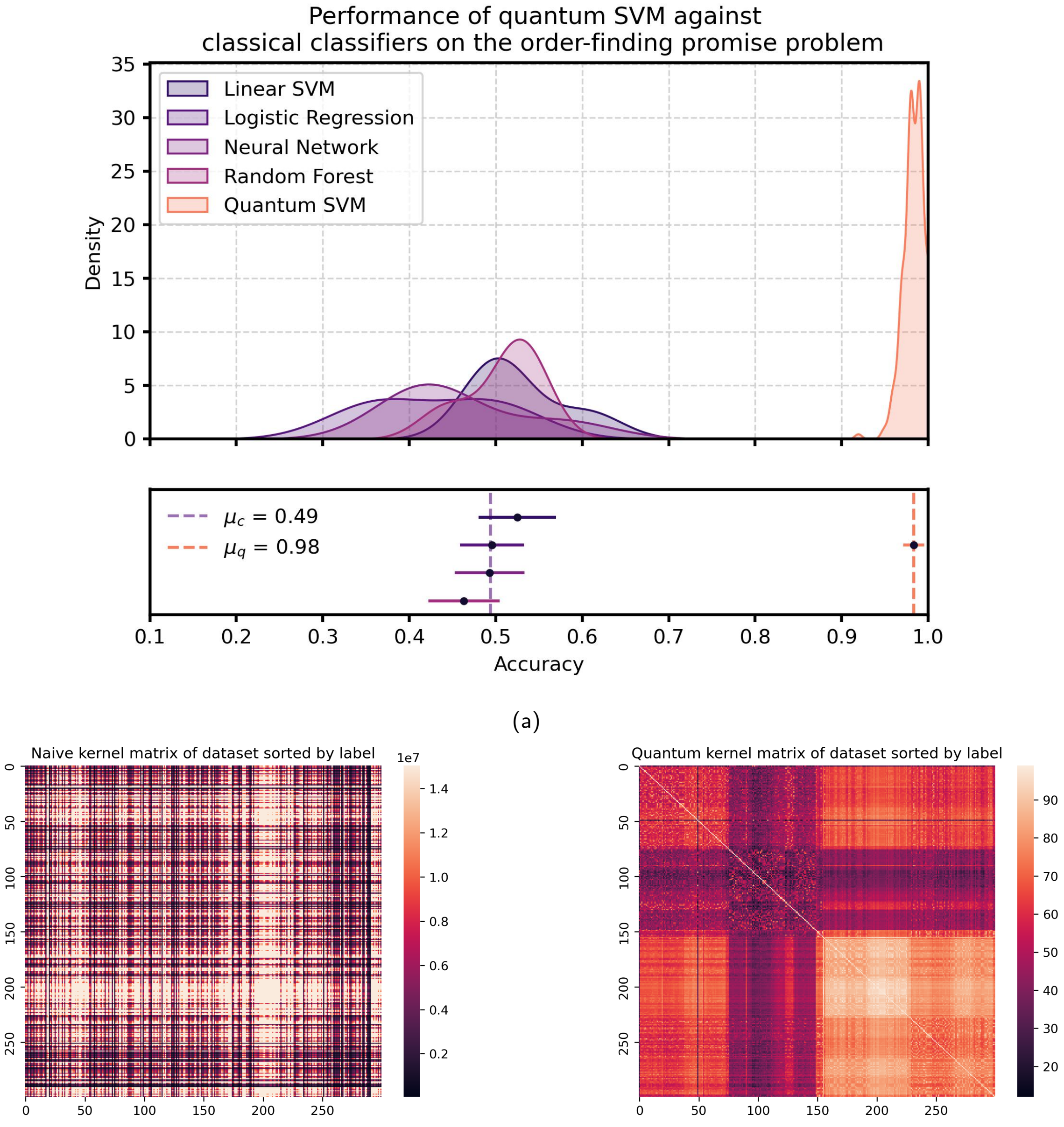
In Figure 6a, we see the distribution of accuracies obtained by classical machine learning models compared to that of the `QKE-SVM` in classifying data points from the order-finding promise problem Π^{OFP}. Observe that the classical models collectively achieve an average accuracy of $\mu_c = 49\%$. Since the dataset used for training had equal proportions of NO and NO instances, this indicates that the classical models performed no better than mere random guessing, suggesting the classical intractability of the Π^{OFP}. In stark contrast to this, the `QKE-SVM` attains a near perfect average accuracy of $\mu_c = 98\%$ and with much lower variance than the classical models. The fact that the `QKE-SVM` has been able to efficiently solve the Π^{OFP} to such high accuracy demonstrates quantum speedup as one could imagine the classical models requiring exponentially more training instances to be able to perform comparably well.

To see why there is such a wide gap in performance, it is instructive to compare the `QKE-SVM` with its classical counterpart, the `Linear-SVM`. Figures 6b and 6c are the kernel matrices used by the `Linear-SVM` and `QKE-SVM` respectively. The instances are sorted with respect to labels, so that the first 150 instances are labelled NO, and the second half are labelled NO. Recall that each entry corresponds to an inner product $K_{\mathcal{V}}(N_1, N_2)$ between two semprimes N_1 and N_2 as represented in some feature space $\mathcal{V}$. For the linear SVM, this feature space is merely the original space, and $K_{\text{classical}}(N_1, N_2) = N_1N_2$. The corresponding kernel matrix 6b reveals no structure that is correlated with the class labels as the cross-hatching patterns appear randomly distributed. From this we can infer that this simple feature map used by the linear SVM cannot linearly separate between the NO and NO instances. Indeed, it is a feature of the Π^{OFP} that no feature map efficiently implementable by a classical computer can linearly separate the dataset (*c.f.* the condition of classical

Algorithm 1 Quantum feature map for Π^{OFP}

```
Input Working register initalised to $|0\rangle$, an instance of the dataset, $n$, which is a large semiprime. Also provided as input is the oracle $U_{\mathcal{P}}$ for checking parity
Output The state $|\phi_n\rangle$ from ??.

1: Generate a uniform superposition over the interval $[0, M]$:
     $|0\rangle \xrightarrow{H^{\otimes\lceil\log(M+1)\rceil}} \sum_{i=0}^{2^{\lceil\log(M+1)\rceil}} |i\rangle \xrightarrow{U_{AA}} \sum_{i=0}^{M} |i\rangle$   (31)
2: Mark integers coprime with $N$ using a similar method to Ref. [? ] for generating a prime state:
     $\sum_{i=0}^{M} |i\rangle \xrightarrow{U_{\text{gcd}}} \sum_{i=0}^{M} |i\rangle |\overbrace{gcd(i \mod N, N)}^{d_i}\rangle$
     $\xrightarrow{U_{=1?}} \sum_{i=0}^{M} |i\rangle |d_i\rangle |\delta_{d_i,1}\rangle$
3: Apply the derandomised algorithm for order-finding conditioned on the third register
     $\xrightarrow{C-U_{\text{ord}}} \sum_{i=0}^{M} |i\rangle |d_i\rangle |\delta_{d_i,1}\rangle |\overbrace{ord_N(i \mod N)}^{o_i}\rangle$   (32)
4: Apply $U_{\mathcal{P}}$ to the last register $o_i$ which calculates the parity of the orders:
     $\xrightarrow{U_{\mathcal{P}}} \sum_{i=0}^{M} |i\rangle |d_i\rangle |\delta_{d_i,1}\rangle |o_i\rangle |C_N(i)\rangle$   (33)
5: Uncompute all but the first and last registers:
     $\xrightarrow{U_{\text{gcd}}^{\dagger} U_{=1?}^{\dagger} C-U_{\text{ord}}^{\dagger}} \sum_{i=0}^{M} |i\rangle |C_N(i)\rangle = |C_N\rangle$   (34)
```



(a)

(b)

(c)

Figure 6: **Smoothed histograms displaying several classical machine learning models being drastically outperformed by the `QKE-SVM` at the task of Π^{OFP} (6a). 6b and 6c respectively show kernel matrices corresponding to the classical and quantum feature maps that were used, revealing that the latter clearly separates instances based on their binary label unlike the former.** The bottom panel of 6a summarises the average accuracy of each model flanked by a σ error bar. The dataset on which the models were trained had an even distribution of NO and NO instances, thus the classical models on average performed no better than random guessing ($\mu_c = 49\%$) suggesting the classical intractability of the Π^{OFP}.

intractability **O.1**). On the other hand, the quantum kernel matrix 6c used by the `QKE-SVM` reveals entry values that are clearly correlated with their class labels. The bright square occupying the entire fourth quadrant in contrast to the relatively darker patch of the first and third quadrants illustrates that in the quantum feature space (**??**), NO instances are more similar than to each other than they are to NO instances. Thus, there exists a wide-margin hyperplane (*c.f.* Equation (**??**)) that linearly separates the dataset. The difference between these two kernel matrices explains why the `QKE-SVM` is able to drastically outperform `Linear-SVM`.

5.6 Summary

We have designed a new classically intractable promise problem based on order finding called Π^{OFP} which admits quantum speedup. We argued that the Π^{OFP} is classically intractable as it is unlikely to be solvable without first performing the classically intractable task of integer factorisation which is equivalent to order finding. We then showed that the Π^{OFP} belongs to SZK and thereby admits a linearly separable representation in some quantum feature space. Then, we implemented the quantum feature map into this space, which was made efficient due to its not requiring index erasure. Through simulations, we evaluated the performance of classical models on the Π^{OFP} against that of `QKE-SVM` and saw that while classical models can perform no better than random guessing, the `QKE-SVM` achieves a near perfect accuracy.

6 Conclusion and outlook

We have provided the only other instance of a classically intractable promise problem, the Π^{OFP}, which can be efficiently solved by a quantum machine learning model, the `QKE-SVM`. More broadly, we have provided a general framework by which new classically intractable promise problems admitting quantum speedup can be constructed.

We obtained this framework by deconstructing Liu *et al.'s* Π^{DLP} and identifying three sufficient properties that render a promise problem Π classically intractable all while solvable by the `QKE-SVM`:

1. Π belongs to the class of Statistical Zero Knowledge (SZK) problems. This induces a quantum feature space in which the promise problem becomes linearly separable.

2. The Quantum Sampling reduction of Π belongs to the class of Exact Quantum Polynomial-time (EQP) algorithms. This ensures the quantum feature map can be implemented efficiently and exactly using a quantum computer.

3. Π reduces to an Abelian Hidden Subgroup Problem (AHSP) which confers the promise problem potential classical intractability[9].

In arriving at Π^{OFP}, we investigated a counterexample based on quadratic residuosity, Π^{QRP}, which despite belonging to SZK and being strongly conjectured to be classically intractable, failed to admit an efficient exact implementation of its corresponding quantum feature map. Along the way, we also provided a quadratic quantum speedup for the Π^{QRP} by designing an algorithm for the task of index erasure.

Future research can be based on constructing more novel promise problems using the framework above. The task of 'engineering' promise problems is admittedly highly contrived, especially when the field of machine learning concerns practical promise problems obtained from the real world, such as tumour classification or image recognition. Therefore, future work should be dedicated to using the framework above as a diagnostic tool for identifying *real world* — and not merely mathematical — datasets that can uniquely be solved by quantum computers.

7 Acknowledgements

We extend our heartfelt gratitude to Prof. Charles Hill and Sam Sutherland for their unwavering support and relentless dedication in meticulously verifying the proofs of the theorems presented in Section C and the intricate statistical analysis of Π^{OFP}. Their invaluable contributions have significantly enhanced the quality and rigor of this work.

References

[1] Dorit Aharonov and Amnon Ta-Shma. Adiabatic Quantum State Generation and Statistical Zero Knowledge, January 2003. URL `http://arxiv.org/abs/quant-ph/0301023`. arXiv:quant-ph/0301023.

[2] Iris Cong and Luming Duan. Quantum Discriminant Analysis for Dimensionality Reduction and Classification. *New Journal of Physics*, 18(7):073011, July 2016. ISSN 1367-2630. DOI: 10.1088/1367-2630/18/7/073011. URL `http://arxiv.org/abs/1510.00113`. arXiv:1510.00113 [quant-ph].

[9] unless it reduces to a problem in **P**

[3] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum random access memory. *Physical Review Letters*, 100(16):160501, April 2008. ISSN 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.100.160501. URL http://arxiv.org/abs/0708.1879. arXiv:0708.1879 [quant-ph].

[4] Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567 (7747):209–212, March 2019. ISSN 0028-0836, 1476-4687. DOI: 10.1038/s41586-019-0980-2. URL http://www.nature.com/articles/s41586-019-0980-2.

[5] Jonas Jäger and Roman V. Krems. Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines, July 2022. URL http://arxiv.org/abs/2207.05865. arXiv:2207.05865 [quant-ph].

[6] Yunchao Liu, Srinivasan Arunachalam, and Kristan Temme. A rigorous and robust quantum speed-up in supervised machine learning. *Nature Physics*, 17(9):1013–1017, September 2021. ISSN 1745-2473, 1745-2481. DOI: 10.1038/s41567-021-01287-z. URL https://www.nature.com/articles/s41567-021-01287-z.

[7] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9):631–633, September 2014. ISSN 1745-2473, 1745-2481. DOI: 10.1038/nphys3029. URL http://arxiv.org/abs/1307.0401. arXiv:1307.0401 [quant-ph].

[8] Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran. Quantum embeddings for machine learning, February 2020. URL http://arxiv.org/abs/2001.03622. arXiv:2001.03622 [quant-ph].

[9] Apimuk Sornsaeng, Ninnat Dangniam, Pantita Palittapongarnpim, and Thiparat Chotibut. Quantum diffusion map for nonlinear dimensionality reduction. *Physical Review A*, 104(5):052410, November 2021. ISSN 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.104.052410. URL http://arxiv.org/abs/2106.07302. arXiv:2106.07302 [quant-ph].

A Algorithm for index erasure with quadratic speedup

Algorithm 2 Index Erasure

```
Input The initial state $\sum_{i\in X}|i\rangle f(i)$. An oracle $U_f$ with action $|i\rangle|0\rangle \xrightarrow{U_f} |i\rangle|f(i)\rangle$ implementing a one-way function $f : X \to Y$ that is M-to-one. An oracle $|x\rangle|y\rangle|q\rangle \xrightarrow{U_g} |x\rangle|y\rangle|q \oplus g_f(x,y)\rangle$ implementing the verifier $g_f(x,y) = [y = f(x)]$. A unitary operator $U_X$ producing a uniform superposition over the input states.
Output The state $\sum_{i\in X}|f(i)\rangle$.
Note For clarity, the algorithm will begin with the output first from which the initial state will be reversibly reconstructed.
1:
    $\sum_{i\in X}|f(i)\rangle|0\rangle \xrightarrow{U_X} \sum_{i\in X}\sum_{j\in X}|f(i)\rangle|j\rangle$
2: Up to the addition and removal of a third register set to $|+\rangle$, obtain:
    $\xrightarrow{U_g} \sum_{i\in X}\sum_{j\in X}(-1)^{g_f(f(i),j)}|f(i)\rangle|j\rangle$
   Note that out of the $|X||Y|$ states, $|X|$ will be 'marked' with a negative phase, so the success probability of their measurement is $1/|Y| = M/|X|$.
3: Perform amplitude amplification on the previous state:
    $\xrightarrow{U_{AA}} \sum_{i\in X}|f(i)\rangle|i\rangle$
4: Note that following the steps (1)-(3) but in reverse yields the desired operation $U_{UNCOMP} := (U_X U_g U_{AA})^\dagger$.
```

A.1 Index erasure for implementing the quantum feature map in the quadratic residuosity promise problem Π^{OF}

We can use Algorithm 2 to solve the state generation problem for the quadratic residuosity promise problem with quadratic speedup, as shown in 3.

B Proof of Theorem 4

Proof. We begin by first highlighting that $\mathbb{Z}_n^*$ is

- Finite as $|\mathbb{Z}_n^*| = \varphi(n)$ which is well-defined; and
- Abelian, as modular multiplication is commutative: $a \times b \mod n = b \times a \mod n$.

Since $\mathbb{Z}_n^*$ is a finite abelian group, by the Converse Lagrange Theorem (CLT), for all $k \,|\, \varphi(n)$, there exists at least one subgroup $H \leq \mathbb{Z}_n^*$ whose order is exactly k. In particular, since $m|\varphi(n) = 2^a m$, then there exists a subgroup H of order m. Now suffice it to prove that

$$x \in \mathbb{Z}_n^* \text{ has an odd order } \iff x \in H \leq \mathbb{Z}_n^* \text{ where } |H| = m, \text{ which exists by the CLT} \tag{39}$$

$\Leftarrow$ If $x \in H$, then the subgroup $\langle x\rangle$ generated by x must also be contained in H:

$$x \in H \implies \langle x\rangle \leq H \tag{40}$$

Algorithm 3 Quadratic Residuosity state generation

Input Working register initalised to $|0\rangle$, an element $x \in \mathbb{Z}_N^*$, and the quantum oracle U_f implementing $f(x) = x^2 \mod N$.

Output The state $|C_x\rangle$.

1: Generate a uniform superposition over the elements of $\mathbb{Z}_N^*$ using the method of Ref. [?] for generating a prime state:

$$|0\rangle \xrightarrow{H^{\otimes n}} \sum_{i=0}^{N-1} |i\rangle \xrightarrow{U_{\text{gcd}}} \sum_{i=0}^{N-1} |i\rangle \, |gcd(i,N)\rangle \xrightarrow{U_{\text{AA}}} \sum_{i\in\mathbb{Z}_N^*} |i\rangle \, |1\rangle \equiv \sum_{i\in\mathbb{Z}_N^*} |i\rangle \tag{35}$$

Note that amplitude amplification is possible because the success probability of i and N being coprime is known a priori to be $\phi(N)/N = (p-1)(q-1)/pq \approx 1$ for large p, q [?].

2:

$$\xrightarrow{U_f} \sum_{i\in\mathbb{Z}_N^*} |i\rangle \, |i^2 \mod N\rangle \tag{36}$$

3: Using the technique of Algorithm 2

$$\xrightarrow{U_{\text{UNCOMP}}} \sum_{i\in X} |i^2 \mod N\rangle \tag{37}$$

4: Modular multiplication given by

$$\xrightarrow{U_\times} \sum_{i\in X} |i^2 \cdot x \mod N\rangle = |C_x\rangle \tag{38}$$

The number of elements in the $\langle x\rangle$ is precisely the order of x, $ord_n(x)$. By Lagrange's theorem, this generated subgroup must divide the group H

$$|\langle x\rangle| = ord_n(x) \mid |H| = m \tag{41}$$

But since m is odd, so must $ord_n(x)$ [?].

$\Rightarrow$ From the Chinese Remainder Theorem,

$$\mathbb{Z}_n^* \cong \mathbb{Z}_p^* \times \mathbb{Z}_q^*$$

Since p and q are prime numbers,

$$\cong C_{\varphi(p)} \times C_{\varphi(q)} \cong C_{p-1} \times C_{q-1}$$

By the fundamental theorem of arithmetic, rewrite $p-1$ and $q-1$ in terms of their prime factorisations $p-1 = 2^{a_1}p_1^{k_1}...p_i^{k_i}$, $q-1 = 2^{a_2}q_1^{l_1}...q_j^{l_j}$ so that

$$\varphi(n) = (p-1)(q-1) = (2^{a_1}p_1^{k_1}...p_i^{k_i})(2^{a_2}q_1^{l_1}...q_j^{l_j}) \tag{42}$$

$$= 2^{a_1}2^{a_2}p_1^{k_1}...p_i^{k_i}q_1^{l_1}...q_j^{l_j} \overset{!}{=} 2^a m \tag{43}$$

It is a well-known fact that $C_{rs} \cong C_r \times C_s$ for coprime r and s. Using this result along with the factorisation in 43, we can further simplify 42 as

$$\begin{aligned} C_{p-1} \times C_{q-1} &\cong C_{2^{a_1}p_1^{k_1}...p_i^{k_i}} \times C_{2^{a_2}q_1^{l_1}...q_j^{l_j}} \\ &\cong C_{2^{a_1}} \times C_{p_1^{k_1}} ... \times ...C_{p_i^{k_i}} \times C_{2^{a_2}} \times C_{q_1^{l_1}} ... \times ...C_{q_j^{l_j}} \end{aligned}$$

which up to reordering is

$$\cong C_{2^{a_1}} \times C_{2^{a_2}} \times C_{p_1^{k_1}} \ldots \times \ldots C_{p_i^{k_i}} \times C_{q_1^{l_1}} \ldots \times \ldots C_{q_j^{l_j}} \, .$$

Note that a cyclic group C_{2^k} only has elements whose orders divide 2^k and thus are even, except for the trivial element $1 \in C_1 \leq C_{2^k}$:

$$x \in C_{2^k} \text{ has odd order } \iff x = 1 \in C_1 \leq C_{2^k} \tag{44}$$

So if $x \in \mathbb{Z}_n^*$ has odd order, then

$$x \overset{!}{\in} C_1 \times C_1 \times C_{p_1^{k_1}} \ldots \times \ldots C_{p_i^{k_i}} \times C_{q_1^{l_1}} \ldots \times \ldots C_{q_j^{l_j}} \leq \mathbb{Z}_n^* \, . \tag{45}$$

But observe that the second term is a subgroup of order $1 \cdot 1 \cdot p_1^{k_1} \ldots p_i^{k_i} q_1^{l_1} \ldots q_j^{l_j} = m$, which is precisely the subgroup H

From the above proof, it follows that of the $\varphi(n)$ elements in $\mathbb{Z}_n^*$ there are $m-1$ elements with odd orders, excluding the trivial element 1 which has order 1. Thus, the proportion of such elements is

$$\frac{m-1}{\varphi(n)} = \frac{m-1}{2^a m} = \frac{1}{2^a}(1 - \frac{1}{m}) \, . \tag{46}$$

□

C Mathematical properties of OF_P^2

Suppose $N = pq \in \mathbb{Z}$ is a large semiprime. In OF_P^2 , an instance N is labeled according to the rule:

$$h(N) = \begin{cases} +1, & \mathcal{E}(\phi(N)) = 2 \\ -1, & \mathcal{E}(\phi(N)) \geq 9 \end{cases} \tag{47}$$

Put simply, h labels all odd semiprimes N whose corresponding totient $\varphi(N)$ is promised to attain an evenness of exactly 2, in which case it is labelled +1, or an evenness of ≥ 9, in which case it is labelled -1. Note that 2 is the minimum evenness that $\varphi(N)$ can attain for a semiprime N since

$$N = pq \implies \varphi(N) = (p-1)(q-1)$$

since N is odd, so are p, q and thus $p-1$, $q-1$ are even

$$\implies \exists k_1, k_2 \in \mathbb{Z} \; \varphi(N) = (2k_1)(2k_2) = 2^2(k_1k_2) = 2^{2+a}m, \; a \in \mathbb{Z}_{\geq 0} \text{ and } m \text{ odd}$$
$$\implies \mathcal{E}(\varphi(N)) = 2 + a \geq 2$$

So, the +1-labelled semiprimes are those whose $\varphi(N)$ has a minimum evenness (of 2).

As a promise problem, the OF_P^2 is formulated as the language Π:

$$\begin{aligned} \Pi &= (\Pi_Y, \Pi_N) \\ \Pi_Y &= \{N \; : \; h(N) = +1\} \\ \Pi_N &= \{N \; : \; h(N) = -1\} \end{aligned} \tag{47}$$

C.1 Conjectured classical intractability

If $\varphi(N)$ is known a priori then $\mathcal{E}(\varphi(N))$ can be efficiently determined in $\mathcal{O}(\log N)$ steps by repeatedly halving $\varphi(N)$ until a non-integer result is obtained. It is also well known that computing $\varphi(N)$ given N is equivalent to factorising N itself. As such, the task of computing the evenness of $\varphi(N)$ has the following reduction:

$$\begin{aligned}\text{Compute } \mathcal{E}(\varphi(N)) &\xrightarrow{\text{Reduces to}} \text{Compute } \varphi(N) = (p-1)(q-1) \\ &\xrightarrow{\text{Equivalent to}} \text{Factorise } N = pq.\end{aligned}$$

From this we can conclude that OF_P^2 is *at most* as hard as factorising integers, which is classically difficult. If it is in fact *as hard*, then OF_P^2 is also classically intractable. However, it is much likelier that it is *easier* as $\mathcal{E}(\varphi(N))$ only yields the exponent of 2 in the prime factorisation of $\varphi(N)$ and not that of any other prime factors. Still, it is not known whether the evenness of $\varphi(N)$ can be classically calculated without explicitly requiring $\varphi(N)$ itself. In light of this, we conjecture as to the classically intractability of the OF_P^2.

C.2 OF_P^2 is an instance of SZK

This promise problem over semiprimes is actually a specific instance of Statistical Difference or $\text{SD}_{\alpha\beta}$ which you will recall belongs to SZK. We now provide some groundwork before we prove this claim.

C.2.1 Preliminaries

Fix a semiprime integer N whose totient has the partial factorisation $\phi(N) = 2^a m$ where m is an odd integer and $a \geq 2$. To show that $OF_P^2 \in \text{SD}_{\alpha\beta}$, we will show that there exists a function parameterised by semiprimes whose image for a fixed semiprime N varies drastically depending on whether that semiprime is a $+1$ or -1 instance.

This function will make use of an important quantity, $P\big(ord_N(i)\big)$ which calculates the parity of the order of a number i with respect to a semiprime $N \geq i$. If i is not coprime with N, we use the convention that $ord_N(i) = P(ord_N(i)) = 0$. Given that this quantity can only take three possible values, we will at times use a colour analogy denoting $\{-1, 0, +1\}$ with the colours Red, Green and Blue respectively.

Having defined $P\big(ord_N(i)\big)$, we now introduce the function $\bar{C}_N$ which will bring us one step closer to the final function C_N:

$$\bar{C}_N : [1, N] \to [1, N] \times \{-1, 0, +1\}, \quad i \mapsto \big(i, P\big(ord_N(i)\big)\big) \tag{48}$$

In our proof, we will ultimately care about the global properties of $\bar{C}_N$. In particular, we will require the number of elements in the domain of $\bar{C}_N$ that map to the same value. This can be calculated using our earlier analysis in **??**:

$$\bar{C}_N(i) = \begin{cases} (i, \quad 0) & \text{for } G := N - \varphi(N) & \text{elements} \\ (i, +1) & \text{for } B := \varphi(N)(1 - \frac{1}{2^a}) & \text{elements} \\ (i, -1) & \text{for } R := \varphi(N)(\frac{1}{2^a}) & \text{elements} \end{cases} \tag{49}$$

For reasons that will be explained later, we will make a slight change to $\bar{C}_N$ by extending its domain. Suppose it is known that the largest semiprime in the domain of our labeling rule h is $N_{\max}$ and fix a large multiple $M = kN_{\max} > N$ for some large positive integer k, the choice of which will be discussed later. Now consider the function

$$C_N : [1, M] \to [1, M] \times \{-1, 0, +1\}, \quad i \mapsto \big(i, P\big(ord_N(i \mod N)\big)\big) \tag{50}$$

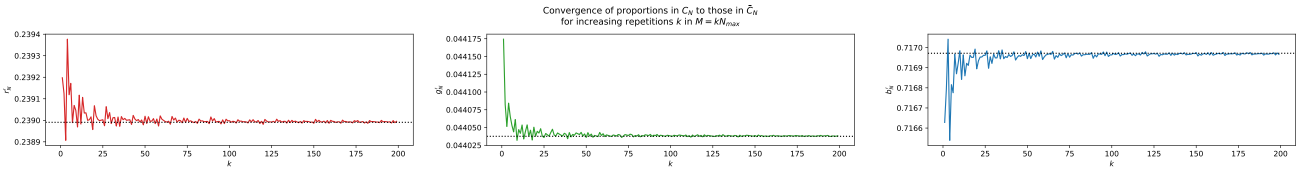


Figure 7: **Convergence of dashed proportions c' of marble colours to their undashed counterparts c with increasing repetitions k.**

which is essentially a periodic version of $\bar{C}_N$. Specifically, if $M = m_N N + r$ for positive integers m_N and r_N, the image of C_N is that of $\bar{C}_N$ repeated in full m_N times across the interval $[1, m_N N]$ as well as one more round extending only over the first r_N elements. It will be important to track how this change will affect the global properties of the function. It is evident the distribution of elements calculated for the aperiodic version in Equation 49 will only be multiplied m_N times along with some remainder:

$$C_N(i) = \begin{cases} (i, \quad 0) & \text{for } G' \coloneqq m_N G + r_G \quad \text{elements} \\ (i, +1) & \text{for } B' \coloneqq m_N B + r_B \quad \text{elements} \\ (i, -1) & \text{for } R' \coloneqq m_N R + r_R \quad \text{elements} \end{cases} \tag{51}$$

where $r_N = r_R + r_B + r_G < R + B + G = N$. While it may seem like the global properties of the two functions differ greatly, because we will care most about their relative proportions we find that k and by extension m_N can be chosen such that the differences become asymptotically small. For example, the proportion of green (0) instances in C_N and $\bar{C}_N$ differs by

$$\begin{aligned} \left| \frac{G}{N} - \frac{m_N G + r_G}{m_N N + r_N} \right| &= \left| \frac{G}{N} - \left(\frac{G}{N} - \frac{r_N G/N - r_G}{m_N N + r_N} \right) \right| \\ &= \left| \frac{r_N G/N - r_G}{m_N N + r_N} \right| \\ &\leq \begin{cases} \frac{r_N G/N}{m_N N} \leq \frac{r_N}{m_N N} & r_G < 2 r_N G/N \\ \frac{r_G}{m_N N} & r_G > 2 r_N G/N \end{cases} \\ &\leq \frac{1}{m_N} \leq \frac{1}{k} \end{aligned}$$

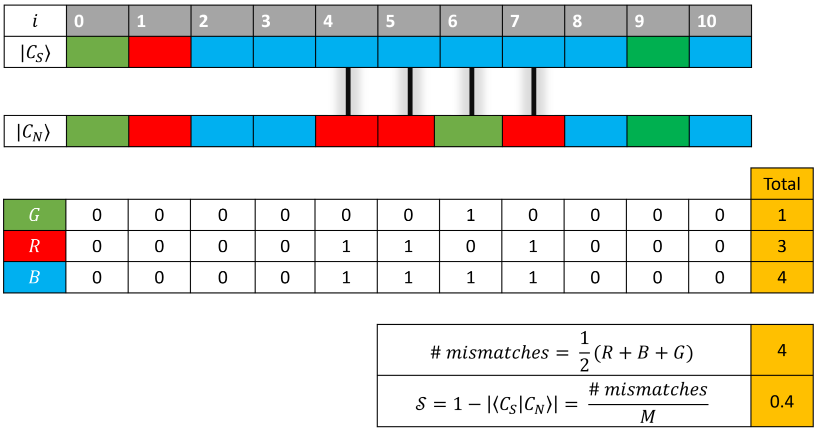
Thus, given ϵ error, it suffices to select $k > \frac{1}{\epsilon}$ and $k = \mathcal{O}(\frac{1}{\epsilon})$. From now on, we will assume that the proportion of the domain of C_N mapping to the same colour is at most ϵ away from that of $\bar{C}_N$: for example, $\left| \frac{G}{N} - \frac{G'}{M} \right| = \epsilon$. This relationship is demonstrated for a specific instance if Figure 8.

In our discussion so far of proportions of domains that map to the same value we have implicitly made references to the concept of the probability distribution D_f induced by a function f, which is defined as

$$D_f : \mathrm{Im}(f) \to [0,1], \quad x \mapsto \frac{|f^{-1}(\{x\})|}{|\mathrm{dom}(f)|} \tag{52}$$

In words, $D_f(x)$ refers to the proportion of inputs i in the domain of f whose image is $f(i) = x$. $\mathrm{SD}_{\alpha\beta}$ is the promise problem defined as

$$\begin{aligned} \Gamma &= (\Gamma_{\mathrm{Y}}, \Gamma_{\mathrm{N}}) \\ \Gamma_{\mathrm{Y}} &= \{(f,g) \ : \ |D_f - D_g| > \alpha\} \\ \Gamma_{\mathrm{N}} &= \{(f,g) \ : \ |D_f - D_g| < \beta\} \end{aligned} \tag{53}$$



i	0	1	2	3	4	5	6	7	8	9	10
$\vert C_S\rangle$	🟩	🟥	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟩	🟦
$\vert C_N\rangle$	🟩	🟥	🟦	🟦	🟥	🟥	🟩	🟥	🟦	🟩	🟦

												Total
G	0	0	0	0	0	0	1	0	0	0	0	1
R	0	0	0	0	1	1	0	1	0	0	0	3
B	0	0	0	0	1	1	1	1	0	0	0	4

$\#\, mismatches = \frac{1}{2}(R+B+G)$	4
$\mathcal{S} = 1 - \vert\langle C_S \vert C_N\rangle\vert = \frac{\#\, mismatches}{M}$	0.4

Figure 8: **Calculation of the statistical difference between the order spectrum of two semiprimes literally corresponds to counting the number of colour mismatches between those two arrays.**

where $0 \leq \beta \leq \alpha \leq 1$ satisfying $\alpha^2 \geq \beta$.

Here, the statistical difference is defined as:

$$|D_f - D_g| = \frac{1}{2} \sum_{i \in \mathrm{Im}(f)} |D_f(i) - D_g(i)| \tag{54}$$

Our general approach will consist of proving that the OF_P^2 Karp-reduces to an instance of $\mathrm{SD}_{\alpha\beta}$, which is an SZK-complete problem and thus belongs to SZKitself. We now apply the definition of $\mathrm{SD}_{\alpha\beta}$ to our function C_N which was introduced in Equation 50. Suppose we have two semiprimes N and S belonging to the domain of h. The statistical difference between C_N and C_S can be written in the following way:

$$\begin{aligned}
|D_{C_N} - D_{C_S}| &= \frac{1}{2}\sum_{i=1}^{M}\sum_{j\in\{-1,0,+1\}} |D_{C_N}((i,j)) - D_{C_S}((i,j))| \\
&= \frac{1}{2}\sum_{i=1}^{M}\sum_{j\in\{-1,0,+1\}} \left| \frac{|C_N^{-1}(\{(i,j)\})|}{|\mathrm{dom}(C_N)|} - \frac{|C_S^{-1}(\{(i,j)\})|}{|\mathrm{dom}(C_S)|} \right| \\
&= \frac{1}{2M}\sum_{j\in\{-1,0,+1\}} \underbrace{\sum_{i=1}^{M} ||C_N^{-1}(\{(i,j)\})| - |C_S^{-1}(\{(i,j)\})||}_{X(j)}
\end{aligned}$$

$$\mathcal{S} = \frac{1}{2M}\big(\underbrace{X(0)}_{:=\mathcal{G}} + \underbrace{X(+1)}_{:=\mathcal{B}} + \underbrace{X(-1)}_{:=\mathcal{R}}\big)$$

The variables $X(j)$ have quite a visual interpretation. Consider M marbles stacked in a column with the marble in the i^{th} position labelled either red, green, or blue according to the value of $P(ord_N(i \mod N))$. We arrange two of these columns abreast, the marbles of one coloured according to $P(ord_N(i \mod N))$ and those of the other $P(ord_S(i \mod S))$. In this setup, $X(j)$ represents the number of pairs of adjacent marbles across the two columns where only one is coloured j. This is depicted in Figure ??. In fact, together with the factor of $1/2M$, $\mathcal{S}$ represents the total number of pairs of adjacent marbles with mismatched colours. This interpretation is depicted in figure ??.

For the proof, we will obtain upper and lower bounds for $\mathcal{S}$ that will end up satisfying the definition of $\mathrm{SD}_{\alpha\beta}$. After the proof, we also adopt a probabilistic approach to give us further insights into the value of $\mathcal{S}$ on average.

We are now ready to begin the proof.

C.2.2 The proof

Definition C.1 (C-(mis)match)**.** A pair of coloured marbles is called a C-mismatch if only one marble from the pair is coloured C. Otherwise if both marbles are coloured C then the pair is called a C-match.

The proof will make use of the following lemma which places bounds on the total number of C-mismatches between two rows of marbles.

Lemma 6. Consider two rows of M coloured marbles that contain C_1 and C_2 C-coloured marbles respectively. The total number of C-mismatches between the two rows is bounded by the range

$$[|C_1 - C_2|, M - |C_1 + C_2 - M|] \tag{55}$$

Proof. To obtain the lower bound, it is clear that the case where there is a minimum number of C-mismatches corresponds to the one where there is a maximum number of C-matches. There can at most be $\min(C_1, C_2)$ C-matches between the two rows, in which case there is a remaining $\max(C_1, C_2) - \min(C_1, C_2) = |C_1 - C_2|$ C-mismatches left behind.

For the upper bound, we need to minimise the number of C-matches. One way of doing this is to place the C_1 C-coloured marbles at the left end of the first row, and the C_2 C-coloured marbles at the right end of the second row. In the case where $C_1 + C_2 \leq N$, the tails of the two chains will never overlap, creating a maximum of $C_1 + C_2$ C-mismatches. However, if $C_1 + C_2 > N$, there will be an overlap giving rise to $C_1 - (M - C_2) = C_1 + C_2 - M$ C-matches. We must deduct twice this value from $C_1 + C_2$ to get the correct number of C-mismatches. We can combine both of these cases in a single expression:

$$\begin{aligned} C_1 + C_2 - 2\max(C_1 + C_2 - M, 0) &= \min(2N - (C_1 + C_2), \; C_1 + C_2) \\ &= M - |C_1 + C_2 - M| \end{aligned}$$

where the factor of two accounts for the double-counting of C-mismatches in the $C_1 + C_2$ term. □

Theorem 7. *The order-finding promise problem OF_P^2 outlined in Equation 47 reduces to* $\mathrm{SD}_{\alpha\beta}$ *for all semiprimes N with $\frac{\varphi(N)}{N} > 99.6\%$.*

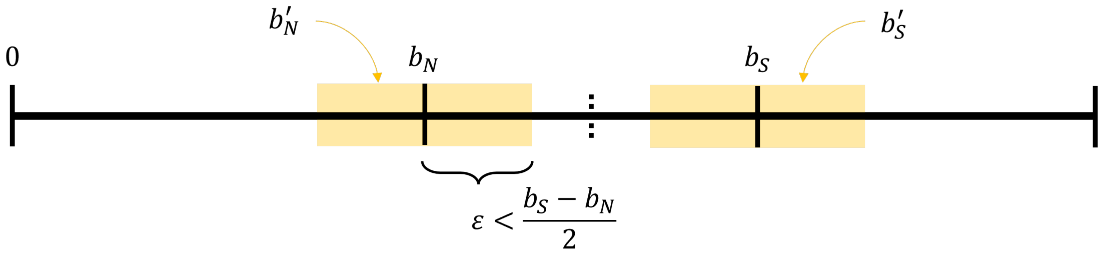
Proof. To present a Karp reduction, we have to show that there exists a polynomial-time, computable function f mapping from OF_P^2 to $\mathrm{SD}_{\alpha\beta}$ as the promise problems that we defined them to be in 47 and 53, such that

$$\begin{aligned} \forall N \in \Pi_\mathrm{Y} &\implies f(N) \in \Gamma_\mathrm{Y} \\ \forall N \in \Pi_\mathrm{N} &\implies f(N) \in \Gamma_\mathrm{N} \end{aligned}$$

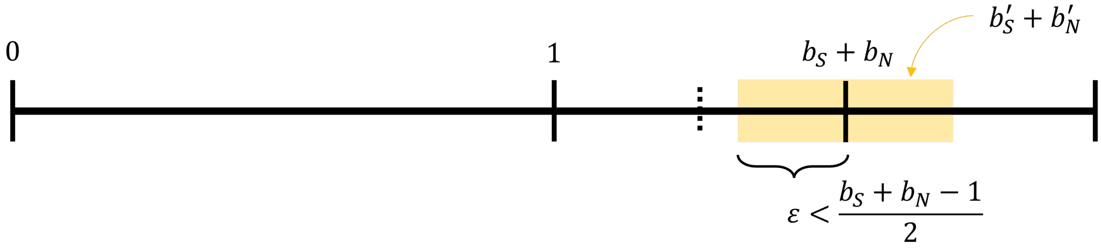
The above condition means that under an input N, f must efficiently produce two functions g_1 and g_2 which have the required statistical differences of $\geq \alpha$ and $\leq \beta$ for when $h(N) = +1$ and -1 respectively for some $0 \leq \beta \leq \alpha \leq 1$. As also stipulated in definition 53 of $\mathrm{SD}_{\alpha\beta}$, we must additionally show that $\alpha^2 \geq \beta$. We will show that the function $f(N) = (C_N, C_S)$ with C_i defined in Equation 51 both satisfies this condition and is efficiently computable, provided S is any fixed semiprime with $h(S) = -1$. In our proof, we additionally assume that we are dealing with sufficiently large semiprimes N, S such that $\frac{\varphi(N)}{N}, \frac{\varphi(S)}{S} > \frac{2^\beta - 1}{2^\beta} > 99.6\%$ where $\beta \geq 9$.

Determining α, β satisfying $\alpha^2 > \beta$. As per the definition of $\mathrm{SD}_{\alpha\beta}$, it is clear that the tightest candidates for α and β are

$$\begin{aligned} \alpha &:= \mathcal{S}_+ = \inf\{|D_{C_N} - D_{C_S}| \; : \; h(N) = +1\} \\ \beta &:= \mathcal{S}_- = \sup\{|D_{C_N} - D_{C_S}| \; : \; h(N) = -1\} \end{aligned}$$



(a) Lower bound



(b) Upper bound

Figure 9: **Visual intuition for the selection of ϵ in the proof of Theorem 7.**

Within this notation, our task translates to showing $\mathcal{S}_+^2 > \mathcal{S}_-$. While it may be difficult to evaluate these two values, we note that it suffices to identify bounds $L_+ < \mathcal{S}_+$ and $U_- > S_-$, and show that $L_+^2 > U_-$.

The expression in Equation **??** corresponds to the sum of C-mismatches across the two columns for each of the three different colours of marbles: $\mathcal{R}$, $\mathcal{B}$, and $\mathcal{G}$. Using Lemma 55, the statistical difference can thus be bounded as

$$\mathcal{S} \in \frac{1}{2M}\left[\sum_{C\in\{R,G,B\}} |C'_1 - C'_2|\,, \sum_{C\in\{R,G,B\}} M - |C'_1 + C'_2 - M|\right] \tag{56}$$

$$= \frac{1}{2}\left[\sum_{c\in\{r,g,b\}} |c'_1 - c'_2|\,, \sum_{c\in\{r,g,b\}} 1 - |c'_1 + c'_2 - 1|\right] \tag{57}$$

where in the second line we have replaced total marble count C by proportions thereof in lower case $c' = \frac{C'}{M}$. We remind the reader that the dashed notation highlights quantities describing the population statistics of C_N as introduced in 51.

We can simplify this expression by considering the cases $h(N) = \pm 1$ separately and constraining the relative proportions of the colours.

The lower bound L_+. It is easy to see that the lower bound in 57 can be reduced further in the following way:

$$L_+ = \frac{1}{2}\sum_{c\in\{r,g,b\}} |c'_1 - c'_2| \geq \frac{|b'_N - b'_S|}{2} \tag{58}$$

While the value of $|b'_N - b'_S|$ may vary depending on the choice of N and S, we will show that we can find yet another lower bound which is only a function of the evenness of N and S. To this end, we first focus on the undashed quantity $|b_N - b_S|$. We know from Equation 49 that $b_S - b_N$ can be expressed in terms of the evenness:

$$\begin{aligned}
b_S - b_N &= \frac{\varphi(S)}{S}\left(1-\frac{1}{2^{\mathcal{E}(S)}}\right) - \frac{\varphi(N)}{N}\left(1-\frac{1}{2^{\mathcal{E}(N)}}\right) \\
&\geq \frac{2^\beta - 1}{2^\beta}\left(1-\frac{1}{2^{\mathcal{E}(S)}}\right) - \left(1-\frac{1}{2^{\mathcal{E}(N)}}\right) \quad \text{by minimising } \frac{\varphi(S)}{S} \text{ and maximising } \frac{\varphi(N)}{N} \\
&= \frac{1}{2^{\mathcal{E}(N)}} - \frac{2^\beta - 1}{2^\beta}\frac{1}{2^{\mathcal{E}(S)}} - \frac{1}{2^\beta} \\
&> 0.244 \qquad\qquad \text{where } \beta \geq 8,\ \mathcal{E}(S) = 9,\ \mathcal{E}(N) = 2
\end{aligned}$$

Now we relate this quantity to the dashed quantity $|b'_N - b'_S|$. We will make use of the fact that dashed proportions are at most ϵ away from their undashed counterparts, $|b_i - b'_i| \leq \epsilon$. Let $0 < \epsilon < \frac{b_S - b_N}{q_+}$ for a choice of $q_+ > 2$, then

$$\epsilon < \frac{b_S - b_N}{2} \implies b'_N < b_N + \epsilon < b_S - \epsilon < b'_S$$

Using this fact, we can drop the absolute value in the following calculation:

$$\begin{aligned}
|b'_S - b'_N| = b'_S - b'_N &> b_S - \epsilon - (b_N + \epsilon) > b_S - b_N - 2\epsilon \\
&> b_S - b_N - \frac{2(b_S - b_N)}{q_+} = (b_S - b_N)(1 - 2/q_+) > 0.24(1 - 2/q_+)\,.
\end{aligned}$$

Now we can derive a lower bound for L_+:

$$L_+ \geq \frac{|b'_N - b'_S|}{2} \geq 0.244(1/2 - 1/q_+) > 0.12$$

for $q_+ > 10^4$. Note that constraining q_+ as such places the following bound on ϵ.

$$\frac{b_S - b_N}{q_+} > \frac{0.244}{q_+} = 2.44 \times 10^{-4} > \epsilon$$

Upper bound Similar to the previous section, we need only consider the $h(N) = -1$ instances when evaluating the upper bound U_-. The upper bound in 57 can be further bounded above if we minimise the negative terms $|c'_1 + c'_2 - 1|$. First, we focus on blue marbles $c = b$. Let $|b - b'| < \epsilon = \frac{|b_N + b_S - 1|}{2q_-}$ for some $q_- > 1$. Then by the triangle inequality:

$$|b'_N + b'_S - (b_N + b_S)| \leq |b'_N - b_N| + |b'_S - b_S| \leq \frac{|b_N + b_S - 1|}{q_-} \leq |b_N + b_S - 1|$$

We use this result to advance from the first line to the second of the following derivation:

$$\begin{aligned}
|b'_N + b'_S - 1| &= |b'_N + b'_S - (b_N + b_S) + b_N + b_S - 1| \geq ||b'_N + b'_S - (b_N + b_S)| - |b_N + b_S - 1|| \\
&= |b_N + b_S - 1| - |b'_N + b'_S - (b_N + b_S)| \geq |b_N + b_S - 1| - \frac{|b_N + b_S - 1|}{q_-} \\
&= |b_N + b_S - 1|(1 - 1/q_-)
\end{aligned}$$

By symmetry, note that the identical inequalities hold for $c = g$ and $c = r$ provided $\epsilon < \min_{c \in \{r,g,b\}} |c_N + c_S - 1|/2q_-$. In fact, since $\mathcal{E}(N), \mathcal{E}(S) \geq 9$ ensures that

$$b_S + b_N > 2 \times \frac{2^\beta - 1}{2^\beta}\left(1 - \frac{1}{2^9}\right) > 1.98$$

the following is guaranteed:

$$\begin{aligned}
&0 < b_N + b_S - 1 \leq 1 - r_N - g_N - r_S - g_S \leq \min(1-(r_N+r_S), 1-(g_N+g_S)) \\
\implies &|b_N + b_S - 1| < |r_N + r_S - 1|,\ |g_N + g_S - 1|
\end{aligned}$$

And thus epsilon is bounded above by

$$\min_{c\in\{r,g,b\}} \frac{|c_N + c_S - 1|}{2q_-} = \frac{|b_N + b_S - 1|}{2q_-} \geq \frac{0.98}{2q_-} = 4.9 \times 10^{-4} \geq \epsilon \tag{59}$$

With ϵ constrained as above, U_- is upper bounded by

$$\begin{aligned}
U_- &= \frac{1}{2} \sum_{c\in\{r,g,b\}} 1 - |c'_N + c'_S - 1| \\
&< \frac{1}{2}\left(3 - (1 - 1/q_-) \sum_{c\in\{r,g,b\}} |c_N + c_S - 1|\right) \\
&= \frac{1}{2}\left(3 - (1 - 1/q_-)\left(1 - g_N - r_N + b_N - g_S - r_S + b_S\right)\right) \\
&= \left(\frac{3}{2} - \left(1 - \frac{1}{q_-}\right)\left((b_N + b_S) - \frac{1}{2}\right)\right) \\
&< \left(\frac{3}{2} - 1.48\left(1 - \frac{1}{q_-}\right)\right) \leq 0.0132
\end{aligned}$$

for $q_- > 10^4$.

Finally, we note that

$$\alpha^2 \geq L_+^2 \geq 0.12^2 \geq 0.014 > 0.0132 \geq U_- \geq \beta \tag{60}$$

Efficiency of f. There are two underlying tasks to $f(N) = (C_N, C_S)$ which we must show are efficient under the input N: generating a semiprime S such taht $h(S) = -1$ and computing $C_{(\cdot)(i)} \forall i \in [1, M]$. For the former task, observe that a sufficiently small -1 semiprime can be precomputed efficiently and generated in $\mathcal{O}(1)$ time. One such semiprime is the product of the two largest Fermat primes $F_3 = 2^{2^3} + 1$ and $F_4 = 2^{2^4} + 1$ whose totient is

$$\varphi(F_3F_4) = (2^{2^3})(2^{2^4}) = 2^{24} \implies h(F_3F_4) = -1 \tag{61}$$

Furthermore, note that $\frac{\varphi(F_3F_4)}{F_3F_4} > 99.6\%$, as required by our constraint on sufficiently large semiprimes. Thus, $S \leftarrow F_3F_4$ is a sufficient candidate that is efficiently implementable.

As for computing $C_N(i) = (i, P(ord_N(i \mod N)))$ given N, note that the calculation consists of modular arithmetic and parity checking which are both efficiently done. The only nontrivial computation is order finding $ord_N(\cdot)$. If we assume a quantum Turing machine as our model of computation, then order-finding can be done in exact polynomial time using the derandomised order-finding algorithm introduced in REF. This means that the f represents a Karp reduction on a quantum computer. □

Corollary 7.1. *OF_P^2 has a* SZK *proof where the verifier has access to a quantum Turing Machine.*

Proof. Since $\mathrm{SD}_{\alpha\beta}$ is a complete problem for SZK, $\mathrm{SD}_{\alpha\beta} \in$ SZK. By the reduction in Theorem 7, it follows that

$$OF_P^2 \leq_K \mathrm{SD}_{\alpha\beta} \in \textsc{szk} \tag{62}$$

□

To further consolidate the corollary above, it is instructive to see OF_P^2 expressed as an SZK interactive proof. We claim that the following interactive protocol is a statistical zero-knowledge interactive proof:

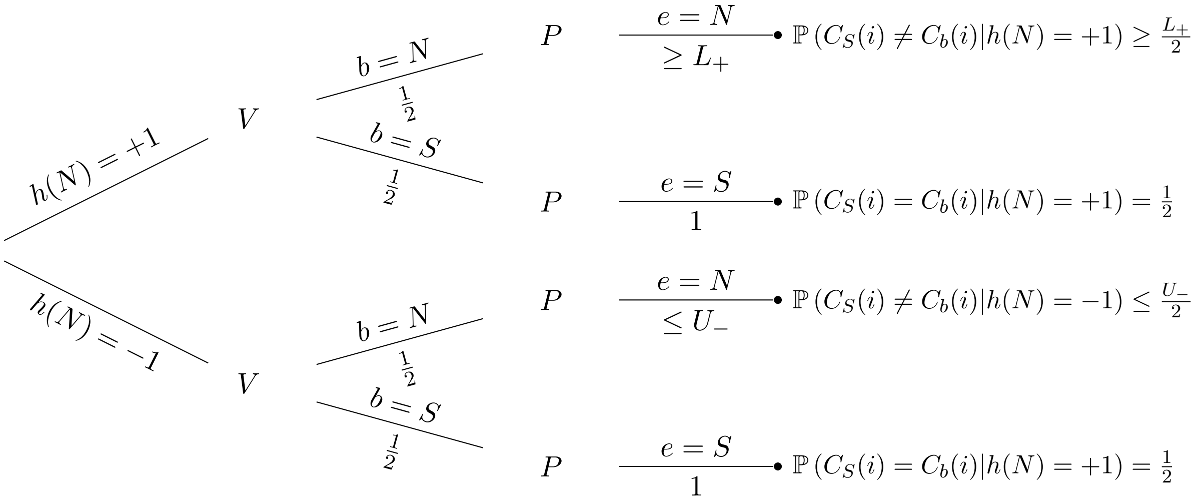
An interactive proof for OF_P^2

Input: Semiprime $N \in \Pi_Y \cup \Pi_N$

Verifier		**Prover**
$S \leftarrow F_3 \times F_4$		
	$\xrightarrow{S}$	
$i \leftarrow\$ [1, M]$ $b \leftarrow\$ \{N, S\}$ $c \leftarrow \big(i, P\big(ord_b(i \mod b)\big)\big)$		
	$\xrightarrow{c}$	
		$d \leftarrow \big(i, P\big(ord_S(i \mod S)\big)\big)$ **if** $d = c$ **then** $e \leftarrow S$ **else** $e \leftarrow N$
	$\xleftarrow{e}$	
if $e = b$ **then** ACCEPT **else** REJECT		

First, we show that the protocol above represents an interactive proof system for the language Π. To do so, we must show that the completeness error ϵ_C (false REJECT) and the soundness error (false ACCEPT) satisfy $1 - \epsilon_C > \epsilon_S + \frac{1}{p(k)}$ for some polynomial $p(k)$ where k is a security parameter.

We can easily obtain these errors through a tree diagram:



$h(N) = +1$
V
$b = N$
$\frac{1}{2}$
$b = S$
$\frac{1}{2}$
P
$e = N$
$\geq L_+$
$\mathbb{P}\left(C_S(i) \neq C_b(i) | h(N) = +1\right) \geq \frac{L_+}{2}$
P
$e = S$
1
$\mathbb{P}\left(C_S(i) = C_b(i) | h(N) = +1\right) = \frac{1}{2}$
$h(N) = -1$
V
$b = N$
$\frac{1}{2}$
$b = S$
$\frac{1}{2}$
P
$e = N$
$\leq U_-$
$\mathbb{P}\left(C_S(i) \neq C_b(i) | h(N) = -1\right) \leq \frac{U_-}{2}$
P
$e = S$
1
$\mathbb{P}\left(C_S(i) = C_b(i) | h(N) = +1\right) = \frac{1}{2}$

Note that only the branches leading to an ACCEPT response by the verifier are shown above. Importantly, the first and third branches refer to outcomes where a mismatch between the value of C_N and C_S has occurred for a randomly chosen $i \leftarrow\$ [1, M]$. The probability of this occurring simply corresponds to the proportion of mismatches among the M pairs of marbles. Indeed, this evaluates exactly to the statistical difference $\mathcal{S}$. This is why the lower and upper bounds derived in the proof of theorem 7 bound the probabilities of the first and third branches respectively. Putting it all together, we get:

$$1 - \epsilon_c = \mathbb{P}(\text{ACCEPT } | h(N) = +1) \geq \frac{1}{2}(1 + L_+) \geq 0.56$$

$$\epsilon_s = \mathbb{P}(\text{ACCEPT } | h(N) = -1) \leq \frac{1}{2}(1 + U_-) \leq 0.51$$

Thus,

$$\implies 1 - \epsilon_C > \epsilon_s + 1/p(k)$$

for a sufficiently large constant polynomial such as $p(k) = 100$.

This shows that the protocol above is an interactive proof system for OF_P^2 . It now remains to show that the proof is also statistical zero knowledge.

An interactive proof is considered statistical zero knowledge if the verifier learns nothing else in their interaction with the prover beyond whether N is a ± 1 instance. This concept is made formal by requiring that the verifier's view of the interaction (i.e. the messages it sends to and receives from the prover) be efficiently and accurately simulated by a simulator that only takes the input string as input. In other words, the simulator must produce a distribution over the message transcripts that is arbitrarily close to that which is obtained in the real interactive proof. Vadhin shows that this requirement is satisfied for all interactive proofs where $L_+^2 > U_-$ (which we showed in theorem 7) by proving a polarization lemma. We refer the interested reader to Vadhin's thesis for more information REF.

C.2.3 Statistical insights

To gain further insight into the value of the statistical difference from Equation **??**, we may reinterpret $\mathcal{S}$ as a probabilistic quantity where $X(j)$ represents a random variable. The justification for this approach is that calculating the preimage $C_N^{-1}(\{(i,j)\})$ requires knowing if i is not coprime with N, or otherwise whether it has an even or odd order. The lack of a closed-form expression for deciding these properties along with our knowledge of the global proportions of C_N is what makes the probabilistic approach preferable.

With $\mathcal{S}$ now a random variable, we will rely on the expectancy $\mathbb{E}[\mathcal{S}]$ to derive characterise the average value of α and β satisfying $\alpha^2 \geq \beta$:

$$\mathbb{E}[\mathcal{S}] = \frac{1}{2M}(\mathbb{E}[\mathcal{R}] + \mathbb{E}[\mathcal{G}] + \mathbb{E}[\mathcal{B}])$$

Focusing just on the Red mismatches as the others follow by symmetry,

$$\mathbb{E}[\mathcal{R}] = \mathbb{E}[\sum_{i=1}^{M} I_i]$$

I_i is an indicator random variable denoting whether the pair of marbles at position i in the column is a $\{R, \neg R\}$ mismatch.

$$\begin{aligned} &= \sum_{i=1}^{M} \mathbb{E}[I_i] = \sum_{i=1}^{M} \mathbb{P}(I_i = 1) \\ &= \#\ \{R, \neg R\} \text{ mismatches} = M \frac{\#\ \{R, \neg R\} \text{ mismatches}}{M} \end{aligned}$$

The fraction on the right is exactly the probability of getting a mismatch on the σ^{th} randomly sampled marble:

$$\begin{aligned} &= M\mathbb{P}(I_\sigma = 1) \\ &= M\big(\mathbb{P}(R_1 \wedge \neg R_2) + \mathbb{P}(\neg R_1 \wedge R_2)\big) \\ &= M\big(\mathbb{P}(R_1)\mathbb{P}(\neg R_2) + \mathbb{P}(\neg R_1)\mathbb{P}(R_2)\big) \\ &= M\big(\mathbb{P}(R_1) + \mathbb{P}(R_2) - 2\mathbb{P}(R_1)\mathbb{P}(R_2)\big) \end{aligned} \tag{63}$$

In this derivation, we have assumed that the colours of pairs of marbles are independent of each other. While this is ultimately a conjecture, we justify this assumption by claiming that a number having an odd order is unlikely to be dependent on another number not having an odd order. Our heuristics indicate that if there is some dependency, then it would have to be marginal and decreasing for increasing semiprimes. Incorporating the result from Equation 63 we get the expectation value:

$$\begin{aligned}\mathbb{E}[\mathcal{S}] &= \frac{1}{2}\Big(\mathbb{P}(R_1)+\mathbb{P}(B_1)+\mathbb{P}(G_1)+\mathbb{P}(R_2)+\mathbb{P}(B_2)+\mathbb{P}(G_2)-2\Big(\mathbb{P}(R_1)\mathbb{P}(R_2)+\mathbb{P}(G_1)\mathbb{P}(G_2)+\mathbb{P}(B_1)\mathbb{P}(B_2)\Big)\Big)\\ &= 1-\mathbb{P}(R_1)\mathbb{P}(R_2)-\mathbb{P}(G_1)\mathbb{P}(G_2)-\mathbb{P}(B_1)\mathbb{P}(B_2)\\ &= 1-\frac{R_1R_2+G_1G_2+B_1B_2}{NS}-\mathcal{O}(\epsilon)\end{aligned}$$

Recalling the proportions derived in Equation 49, we have

$$\begin{aligned}&= 1-\left(\left(1-\frac{\varphi(N)}{N}\right)\left(1-\frac{\varphi(S)}{S}\right)+\frac{\varphi(N)}{N}\frac{\varphi(S)}{S}\left(1-\frac{1}{2^{\mathcal{E}(N)}}\right)\left(1-\frac{1}{2^{\mathcal{E}(S)}}\right)+\frac{\varphi(N)}{N}\frac{\varphi(S)}{S}\left(\frac{1}{2^{\mathcal{E}(S)+\mathcal{E}(N)}})\right)\right)-\mathcal{O}(\epsilon)\\ &= \frac{\varphi(N)}{N}+\frac{\varphi(S)}{S}-\frac{\varphi(S)}{S}\frac{\varphi(N)}{N}\left(2-\frac{1}{2^{\mathcal{E}(N)}}-\frac{1}{2^{\mathcal{E}(S)}}+\frac{1}{2^{\mathcal{E}(N)+\mathcal{E}(S)+1}}\right)-\mathcal{O}(\epsilon)\\ &\overset{S,\,N\to\infty}{=} \frac{1}{2^{\mathcal{E}(N)}}+\frac{1}{2^{\mathcal{E}(S)}}-\frac{1}{2^{\mathcal{E}(N)+\mathcal{E}(S)+1}}-\mathcal{O}(\epsilon)\in\left[\frac{1}{2^{\mathcal{E}(N)}},\ \frac{1}{2^{\mathcal{E}(N)}}+\frac{1}{2^{\mathcal{E}(S)}}\right]\end{aligned}$$

The bounds follow if we ensure that the $\mathcal{O}(\epsilon)<\frac{1}{2^{\mathcal{E}(S)}}-\frac{1}{2^{\mathcal{E}(N)+\mathcal{E}(S)+1}}$ which we can arrange by suitably increasing k in $M=kN_{max}$. If we now fix S such that $h(S)=-1$, then we get

$$\begin{cases}=\left[\frac{1}{4},\frac{9}{32}\right] & h(N)=+1\\ =\left[0,\frac{1}{16}\right] & h(N)=-1\end{cases}$$

Put differently, if we take $\alpha=1/4$ and $\beta=1/16$, the expected statistical difference of -1 and $+1$ semiprimes with respect to D_{C_S} has the following bounds:

$$\begin{aligned}h(N)=+1 &\implies \mathbb{E}[\mathcal{S}]\geq\alpha\\ h(N)=-1 &\implies \mathbb{E}[\mathcal{S}]\leq\beta\,.\end{aligned}$$

By observing that $0\leq\beta\leq\alpha^2\leq 1$, we see that the promise problem defined by h is by definition an instance of $\mathrm{SD}_{\alpha\beta}$ and consequently in SZK*on average*.

This is insufficient for the proof however, as we need to show that we can assign a probability of success of δ to the class membership above that is asymptotically unity. We begin by calculating the variance of the statistical difference:

$$\mathrm{Var}[\mathcal{S}]=\frac{1}{4M^2}(\mathrm{Var}[\mathcal{R}])+\mathrm{Var}[\mathcal{G}]+\mathrm{Var}[\mathcal{B}]+2(\mathrm{Cov}[\mathcal{R},\mathcal{G}]+\mathrm{Cov}[\mathcal{G},\mathcal{B}]+\mathrm{Cov}[\mathcal{B},\mathcal{R}])) \tag{64}$$

Let us focus only on one pair of random variables, as the others can be iterated by symmetry

$$\begin{aligned}\mathrm{Var}[\mathcal{B}+\mathcal{R}] &= \mathrm{Var}[\mathcal{R}])+\mathrm{Var}[\mathcal{B}]+2\mathrm{Cov}[\mathcal{B},\mathcal{R}])\\ &= \mathbb{E}[\mathcal{R}^2]-\mathbb{E}[\mathcal{R}]^2+\mathbb{E}[\mathcal{B}^2]-\mathbb{E}[\mathcal{B}]^2+2\mathbb{E}[\mathcal{R}\mathcal{B}]-2\mathbb{E}[\mathcal{R}]\mathbb{E}[\mathcal{B}]\end{aligned}$$

In this expression there are two types of quantities that we cannot derive by just recycling the expected values derived from the previous step: $\mathbb{E}[\mathcal{R}^2]$ and $\mathbb{E}[\mathcal{R}\mathcal{B}]$. The manner of deriving these quantities will be similar to calculating expected values using indicator random variables

$$\begin{aligned}\mathbb{E}[\mathcal{R}^2] &= \mathbb{E}[\sum_{i,j}I_iI_j]=\sum_{i,j}\mathbb{E}[I_iI_j]=\sum_{i,j}\mathbb{P}(I_i=1\wedge I_j=1\mid i^{\text{th}}\text{ and }j^{\text{th}}\text{ pair of marbles})\\ &= \sum_{i\neq j}\mathbb{P}(I_i=1\wedge I_j=1)+\sum_{i}\mathbb{P}(I_i=1)=\sum_{i\neq j}\mathbb{P}(I_i=1\wedge I_j=1)+\mathbb{E}[\mathcal{R}]\end{aligned}$$

The first term counts the number of distinct pairs of $\{R, \neg R\}$ mismatches along the two columns:

$$= \#\ \{R, \neg R\}\text{—}\{R, \neg R\} \text{ pairs} + \mathbb{E}[\mathcal{R}] = M(M-1)\frac{\#\ \{R, \neg R\}\text{—}\{R, \neg R\} \text{ pairs}}{M(M-1)} + \mathbb{E}[\mathcal{R}]$$

As seen earlier, the fraction is precisely the probability of obtaining two consecutive $\{R, \neg R\}$ mismatches after random sampling without replacement:

$$= M(M-1)\mathbb{P}(I_\sigma = 1 \wedge I_{\sigma+1} = 1 \mid \sigma^{\text{th}} \text{ and } (\sigma+1)^{\text{th}} \text{ pairs randomly sampled}) + \mathbb{E}[\mathcal{R}]$$

There are four combinations of marbles that contribute to this probability term

$$\mathbb{P}(I_\sigma = 1 \wedge I_{\sigma+1} = 1) = \mathbb{P}\left(\begin{matrix}\left\{R \quad \neg R\right\}_\sigma \\ \left\{R \quad \neg R\right\}_{\sigma+1}\end{matrix}\right)$$

For clarity, we will suppress the indices but it should be understood that the bottom row is sampled after the top row.

$$\begin{aligned}
&= \mathbb{P}\left(\begin{matrix}\left(R \quad \neg R\right) \\ \left(R \quad \neg R\right)\end{matrix}\right) + \mathbb{P}\left(\begin{matrix}\left(\neg R \quad R\right) \\ \left(\neg R \quad R\right)\end{matrix}\right) + \mathbb{P}\left(\begin{matrix}\left(\neg R \quad R\right) \\ \left(R \quad \neg R\right)\end{matrix}\right) + \mathbb{P}\left(\begin{matrix}\left(R \quad \neg R\right) \\ \left(\neg R \quad R\right)\end{matrix}\right) \\
&= \mathbb{P}\begin{pmatrix}R \\ R\end{pmatrix}\mathbb{P}\begin{pmatrix}\neg R \\ \neg R\end{pmatrix} + \mathbb{P}\begin{pmatrix}\neg R \\ \neg R\end{pmatrix}\mathbb{P}\begin{pmatrix}R \\ R\end{pmatrix} + \mathbb{P}\begin{pmatrix}\neg R \\ R\end{pmatrix}\mathbb{P}\begin{pmatrix}R \\ \neg R\end{pmatrix} + \mathbb{P}\begin{pmatrix}R \\ \neg R\end{pmatrix}\mathbb{P}\begin{pmatrix}\neg R \\ R\end{pmatrix}
\end{aligned}$$

The probabilities above fall into two main categories:

$$\begin{aligned}
\mathbb{P}\begin{pmatrix}R \\ R\end{pmatrix}\mathbb{P}\begin{pmatrix}\neg R \\ \neg R\end{pmatrix} &= \Big(\frac{R_1'}{M}\frac{R_1'-1}{M-1}\Big)\Big(\frac{M-R_2'}{M}\frac{M-R_2'-1}{M-1}\Big) \\
\mathbb{P}\begin{pmatrix}\neg R \\ R\end{pmatrix}\mathbb{P}\begin{pmatrix}R \\ \neg R\end{pmatrix} &= \Big(\frac{M-R_1'}{M}\frac{R_1'}{M-1}\Big)\Big(\frac{R_2'}{M}\frac{M-R_2'}{M-1}\Big)
\end{aligned}$$

Incorporating these probabilities into the final sum, we get

$$\mathbb{P}\left(\begin{matrix}\left\{R \quad \neg R\right\}_\sigma \\ \left\{R \quad \neg R\right\}_{\sigma+1}\end{matrix}\right) = \frac{\Big(R_1'\left(M-R_2'\right)\Big(-2R_1'R_2' + R_2'\left(M+1\right) + \overline{M}\left(R_1'-1\right)\Big) + R_2'\left(M-R_1'\right)\Big(-2R_1'R_2' + R_1'\left(M+1\right) + \overline{M}\left(R_2'-1\right)\Big)\Big)}{M^2\overline{M}^2}$$

where $\overline{M} = M - 1$. The cross term $\mathbb{E}[\mathcal{B}, \mathcal{R}]$ can similarly be obtained as:

$$\begin{aligned}
\mathbb{E}[\mathcal{B}\mathcal{R}] &= \sum_{i \neq j}\mathbb{P}(I_i^{\mathcal{R}} = 1 \wedge I_j^{\mathcal{B}} = 1) + \sum_i \mathbb{P}(I_i^{\mathcal{R}} = 1 \wedge I_i^{\mathcal{B}} = 1) \\
&= M\overline{M}\mathbb{P}\left(\begin{matrix}\left\{B \quad \neg B\right\}_\sigma \\ \left\{R \quad \neg R\right\}_{\sigma+1}\end{matrix}\right) + M\mathbb{P}(\left\{B \quad R\right\}_\sigma) \\
&= \frac{\left(B_1'R_2' + B_2'R_1'\right)}{M} + \frac{\Big(8B_1'B_2'R_1'R_2' - \overline{M}\left(2B_1R_2' + 2B_2R_1' + \left(B_1' + B_2'\right)\left(-M\left(R_1' + R_2'\right) + 4R_1'R_2'\right) + \left(R_1' + R_2'\right)\left(4B_1'B_2' - M\left(B_1' + B_2'\right)\right)\right)\Big)}{2M\overline{M}}
\end{aligned}$$

Substituting these quantities into Equation 64 and assuming $\overline{M} \approx M$ in the limit of large M, we have for the variance:

$$\begin{aligned}\mathrm{Var}[\mathcal{S}] = &-\frac{{B'_1}^2{B'_2}^2}{M^5} + \frac{3{B'_1}^2B'_2}{4M^4} + \frac{{B'_1}^2B'_2}{4M^5} - \frac{{B'_1}^2}{4M^3} + \frac{3B'_1{B'_2}^2}{4M^4} + \frac{B'_1{B'_2}^2}{4M^5} - \frac{2B'_1B'_2G'_1G'_2}{M^5} + \frac{B'_1B'_2G'_1}{M^4} + \frac{B'_1B'_2G'_2}{M^4}+\\
&\frac{B'_1B'_2R'_1}{M^4} + \frac{B'_1B'_2R'_2}{M^4} - \frac{B'_1B'_2}{M^4} - \frac{2B'_1B'_2R'_1R'_2}{M^5} + \frac{B'_1G'_1G'_2}{M^4} - \frac{B'_1G'_1}{2M^3} - \frac{B'_1G'_2}{2M^3} + \frac{B'_1G'_2}{2M^4} - \frac{B'_1R'_1}{2M^3} - \frac{B'_1R'_2}{2M^3}+\\
&\frac{B'_1}{4M^3} + \frac{B'_1R'_1R'_2}{M^4} + \frac{B'_1R'_2}{2M^4} - \frac{{B'_2}^2}{4M^3} + \frac{B'_2G'_1G'_2}{M^4} - \frac{B'_2G'_1}{2M^3} + \frac{B'_2G'_1}{2M^4} - \frac{B'_2G'_2}{2M^3} - \frac{B'_2R'_1}{2M^3} - \frac{B'_2R'_2}{2M^3} + \frac{B'_2}{4M^3}+\\
&\frac{B'_2R'_1R'_2}{M^4} + \frac{B'_2R'_1}{2M^4} - \frac{{G'_1}^2{G'_2}^2}{M^5} + \frac{3{G'_1}^2G'_2}{4M^4} + \frac{{G'_1}^2G'_2}{4M^5} - \frac{{G'_1}^2}{4M^3} + \frac{3G'_1{G'_2}^2}{4M^4} + \frac{G'_1{G'_2}^2}{4M^5} + \frac{G'_1G'_2R'_1}{M^4} + \frac{G'_1G'_2R'_2}{M^4}-\\
&\frac{G'_1G'_2}{M^4} - \frac{2G'_1G'_2R'_1R'_2}{M^5} - \frac{G'_1R'_1}{2M^3} - \frac{G'_1R'_2}{2M^3} + \frac{G'_1}{4M^3} + \frac{G'_1R'_1R'_2}{M^4} + \frac{G'_1R'_2}{2M^4} - \frac{{G'_2}^2}{4M^3} - \frac{G'_2R'_1}{2M^3} - \frac{G'_2R'_2}{2M^3} + \frac{G'_2}{4M^3}+\\
&\frac{G'_2R'_1R'_2}{M^4} + \frac{G'_2R'_1}{2M^4} - \frac{{R'_1}^2}{4M^3} + \frac{R'_1}{4M^3} - \frac{{R'_2}^2}{4M^3} + \frac{R'_2}{4M^3} + \frac{3{R'_1}^2R'_2}{4M^4} + \frac{3R'_1{R'_2}^2}{4M^4} - \frac{R'_1R'_2}{M^4} - \frac{{R'_1}^2{R'_2}^2}{M^5} + \frac{{R'_1}^2R'_2}{4M^5}+\\
&\frac{R'_1{R'_2}^2}{4M^5}\\
&\leq \frac{9}{M^2} + \frac{33}{M} \leq \frac{42}{M} = \mathcal{O}\left(\frac{1}{M}\right) \implies \sigma\left[\mathcal{S}\right] \leq \sqrt{\frac{42}{M}}
\end{aligned} \tag{65}$$

From $\sigma\left[\mathcal{S}\right]$, we can obtain the success probability δ. Let $\mathcal{S}_\pm$ be the probabilistic statistical difference for $C_{N_\pm}$ where $N_\pm$ are randomly sampled semiprimes wth label ± 1 respectively.

$$\delta = \mathbb{P}\left(\mathcal{S}_- < \mathcal{S}_+^2\right) = \sum_{\substack{s_+, s_- \\ s_+^2 > s_-}} \mathbb{P}\left(\mathcal{S}_- = s_- \wedge \mathcal{S}_+^2 = s_+^2\right)$$

Since N_+ and N_- are randomly sampled, $\mathcal{S}_-$ and $\mathcal{S}_+$ are independent:

$$\begin{aligned}&= \sum_{\substack{s_+, s_- \\ s_+^2 > s_-}} \mathbb{P}\left(\mathcal{S}_- = s_-\right)\mathbb{P}\left(\mathcal{S}_+^2 = s_+^2\right) \geq \sum_{\substack{s_+^2 > \lambda \\ s_- < \lambda}} \mathbb{P}\left(\mathcal{S}_- = s_-\right)\mathbb{P}\left(\mathcal{S}_+^2 = s_+^2\right) \qquad \forall\lambda\\
&= \mathbb{P}\left(\mathcal{S}_- < \lambda\right)\mathbb{P}\left(\mathcal{S}_+^2 > \lambda\right) = \mathbb{P}\left(\mathcal{S}_- < \lambda\right)\mathbb{P}\left(|\mathcal{S}_+| > \sqrt{\lambda}\right)\end{aligned}$$

Let $\mu_\pm = \mathbb{E}[\mathcal{S}_\pm]$ and fix λ so that it satisfies $\lambda > \mu_-$ and $\sqrt{\lambda} < \mu_+$:

$$\begin{aligned}&\geq \mathbb{P}\left(2\mu_- - \lambda < \mathcal{S}_- < \lambda\right)\mathbb{P}\left(\sqrt{\lambda} < \mathcal{S}_+ < 2\mu_+ - \sqrt{\lambda}\right)\\
&= \mathbb{P}\left(|\mathcal{S}_- - \mu_-| < \lambda - \mu_-\right)\mathbb{P}\left(|\mathcal{S}_+ - \mu_+| < \mu_+ - \sqrt{\lambda}\right)\end{aligned}$$

Let $\tilde{\lambda} \coloneqq \min\left(\lambda - \mu_-, \mu_+ - \sqrt{\lambda})\right)$

$$\geq \mathbb{P}\left(|\mathcal{S}_- - \mu_-| < \tilde{\lambda}\right)\mathbb{P}\left(|\mathcal{S}_+ - \mu_+| < \tilde{\lambda}\right)$$

By Chebyshev's inequality

$$\begin{aligned}&\geq \left(1 - \frac{\sigma_+^2}{\tilde{\lambda}^2}\right)\left(1 - \frac{\sigma_-^2}{\tilde{\lambda}^2}\right)\\
1 \geq \delta &\geq \left(1 - \frac{42}{M\tilde{\lambda}}\right)^2 = 1 - \mathcal{O}\left(\frac{1}{M}\right) \to 1 \text{ as } M \to \infty\\
&\therefore \delta \to 1 \text{ as } M \to \infty\end{aligned}$$

Thus, the probability of success approaches unity as M increases, which may be arranged for a large choice of k or otherwise a large maximum semiprime N_{max} in the dataset.