Research Article

Martin Ekerå*

Quantum algorithms for computing general discrete logarithms and orders with tradeoffs

<https://doi.org/10.1515/jmc-2020-0006>

Received Feb 08, 2020; accepted Nov 20, 2020

Abstract: We generalize our earlier works on computing short discrete logarithms with tradeoffs, and bridge them with Seifert's work on computing orders with tradeoffs, and with Shor's groundbreaking works on computing orders and general discrete logarithms. In particular, we enable tradeoffs when computing general discrete logarithms. Compared to Shor's algorithm, this yields a reduction by up to a factor of two in the number of group operations evaluated quantumly in each run, at the expense of having to perform multiple runs. Unlike Shor's algorithm, our algorithm does not require the group order to be known. It simultaneously computes both the order and the logarithm. We analyze the probability distributions induced by our algorithm, and by Shor's and Seifert's order-finding algorithms, describe how these algorithms may be simulated when the solution is known, and estimate the number of runs required for a given minimum success probability when making different tradeoffs.

Keywords: Discrete logarithms, Order finding, Shor's algorithms

2020 Mathematics Subject Classification: 68Q12, 81P68, 94A60

1 Introduction

As in [5, 7, 8], let $\mathbb{G}$ under $\odot$ be a finite cyclic group of order r generated by g , and

$$x = [d]g = \underbrace{g \odot g \odot \cdots \odot g}_{d \text{ times}}.$$

The discrete logarithm problem is to compute $d = \log_g x$ given the group elements g and x . In cryptographic applications, the group $\mathbb{G}$ is typically a subgroup of $\mathbb{F}_p^*$, for some prime p , or an elliptic curve group.

In the general discrete logarithm problem $0 \leq d < r$, whereas d is smaller than r by some order of magnitude in the short discrete logarithm problem.

1.1 Earlier works

In 1994, in a groundbreaking publication, Shor [25, 26] introduced polynomial time quantum algorithms for factoring integers and for computing general discrete logarithms in $\mathbb{F}_p^*$. The latter algorithm may be trivially adapted to compute general discrete logarithms in any finite cyclic group, provided that the group operation can be implemented efficiently quantumly.

Ekerå [5] initiated a line of research in 2016 by introducing a modified version of Shor's algorithm for computing discrete logarithms that more efficiently solves the short discrete logarithm problem. This work

Article note: Funding and support was provided by the Swedish NCSA that is a part of the Swedish Armed Forces.

***Corresponding Author: Martin Ekerå:** KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden; Swedish NCSA, Swedish Armed Forces, SE-107 85 Stockholm, Sweden; Email: ekera@kth.se

is of cryptographic relevance as the short discrete logarithm problem underpins the security of many implementations of cryptosystems instantiated with safe-prime groups. A notable example is Diffie-Hellman key exchange [3] in TLS, IKE and NIST SP 800-56A [2, 9, 13].

In a follow-up work, Ekerå and Håstad [8] enabled tradeoffs in Ekerå's algorithm using ideas that directly parallel those of Seifert [24] in his work on enabling tradeoffs in Shor's order-finding algorithm; the quantum part of Shor's factoring algorithm. Ekerå and Håstad furthermore showed how the RSA integer factoring problem, that underpins the widely deployed RSA cryptosystem [21], may be reduced via [11] to a short discrete logarithm problem and attacked quantumly. This gives rise to a quantum algorithm that more efficiently solves the RSA integer factoring problem when making tradeoffs and comparing to Shor's original factoring algorithm, or to Seifert's factoring algorithm.

Ekerå [7] subsequently refined the classical post-processing in [8] to render it more efficient. With this improved post-processing, the algorithm of Ekerå and Håstad is shown in [7] to outperform Shor's and Seifert's factoring algorithms when targeting RSA integers, irrespective of whether tradeoffs are made.

A key component to this result was the development of a classical simulator for the quantum algorithm for computing short discrete logarithms: For problem instances for which the solution is classically known, this simulator allows outputs to be generated that are representative of outputs that would be generated by the quantum algorithm if executed on a quantum computer. This in turn allows the efficiency of the classical post-processing to be experimentally assessed.

1.2 Our contributions

We generalize and bridge our earlier works on computing short discrete logarithms with tradeoffs, Seifert's work on computing orders with tradeoffs and Shor's groundbreaking works on computing orders and general discrete logarithms. In particular, we enable tradeoffs when computing general discrete logarithms.

Compared to Shor's algorithm for computing general discrete logarithms, this yields a reduction by up to a factor of two in the number of group operations evaluated quantumly in each run, at the expense of having to perform multiple runs. Unlike Shor's algorithm, our algorithm does not require the group order to be known. It simultaneously computes both the order and the logarithm. This allows our algorithm to outperform Shor's original algorithms with respect to the overall number of group operations that need to be evaluated quantumly in some cases even when not making tradeoffs. One cryptographically relevant example of such a case is the computation of discrete logarithms in Schnorr groups of unknown order.

We analyze the probability distributions induced by our algorithm, and by Shor's and Seifert's order-finding algorithms, describe how all of these algorithms may be simulated when the solution to the problem instance is known, and estimate the number of runs required for a given minimum success probability when making different tradeoffs.

1.2.1 On the cryptographic significance of this work

The security of virtually all currently widely deployed asymmetric cryptosystems is based on the intractability of either the discrete logarithm problem or the integer factoring problem.

In this work, we further the understanding of how hard these two key problems are to solve quantumly when not on special form. We hope that our results may prove useful when developing cost estimates for quantum attacks, and that they may inform decisions on when to mandate migration from the currently deployed asymmetric cryptosystems to post-quantum secure cryptosystems.

1.2.2 Further details and overview

Our algorithm for computing general discrete logarithms in turn consists of two algorithms;

- a quantum algorithm, that upon input of a generator g of order r , and an element $x = [d]g$ where $0 \leq d < r$, outputs a pair (j, k) , and
- a classical probabilistic post-processing algorithm, that upon input of a set of n pairs (j, k) , produced by n runs of the quantum algorithm, computes d .

In addition to the above post-processing algorithm, we furthermore specify

- a classical probabilistic post-processing algorithm, that upon input of a set of n integers j computes the order r . Note that the same set of integers j may be used as input to both this and the above post-processing algorithm, by breaking out j from the pairs (j, k) .

The quantum algorithm is identical to the algorithm in [7, 8] for computing short discrete logarithms with tradeoffs. The key difference in this work is that we admit general discrete logarithms and comprehensively analyze the probability distribution that the algorithm induces for such logarithms.

The post-processing algorithm for d is a tweaked version of the lattice-based algorithm in [7], whereas the algorithm for r is a natural generalization of the lattice-based algorithm in [7] first sketched in a pre-print of [8]. It is similar to the post-processing in [24].

The quantum algorithm is parameterized under a tradeoff factor s . This factor controls the tradeoff between the requirements that the algorithm imposes on the quantum computer, and the number of runs, n , required to attain a given minimum probability q of recovering d and r in the classical post-processing.

Following [7], we estimate n for a given problem instance, represented by d and r , and fixed s and q , by simulating the quantum algorithm. We first use simulated output to heuristically estimate n , and then verify the estimate by executing the two post-processing algorithms with respect to simulated output.

The simulator is based on a high-resolution two-dimensional histogram of the probability distribution induced by the quantum algorithm. By sampling the histogram, we generate pairs (j, k) that very closely approximate output that would be produced by the quantum algorithm if executed on a quantum computer.

To construct the histogram, we first derive a closed-form expression that approximates the probability of the quantum algorithm yielding (j, k) as output, and an upper bound on the error in the approximation. We then integrate this expression and the error bound numerically in different regions of the plane.

Our simulations show that when not making tradeoffs, a single run suffices to compute d or r with $\geq 99\%$ success probability. When making tradeoffs, slightly more than s runs are typically required to achieve a similar success probability. In Appendix A we show that these results extend to order finding and factoring.

Note that the simulator requires d and r to be explicitly known: It cannot be used for problem instances represented by group elements g and $x = [d]g$.

1.2.3 Structure of this paper

The quantum algorithm is described in Section 2. In Section 3, we analyze the probability distribution it induces, and derive a closed-form expression that approximates the probability of it yielding (j, k) as output. In Sections 4 and 5, we describe how the high-resolution histogram is constructed by integrating the closed-form expression, and how the histogram is sampled to simulate the quantum algorithm.

In Section 6, we describe the two post-processing algorithms for recovering d and r from a set of n pairs (j, k) . In Section 7, we use the simulator to estimate the number of runs n required to solve a given problem instance for d and r , with minimum success probability q , as a function of the tradeoff factor s .

We summarize past and new results, and discuss related applications, such as order finding and integer factoring, in Sections 8 and 9, and in the appendices.

1.3 Notation

The below notation is used throughout this paper:

- $u \bmod n$ denotes u reduced modulo n constrained to $[0, n)$.
- $\{u\}_n$ denotes u reduced modulo n constrained to $[-n/2, n/2)$.
- $\lceil u \rceil$, $\lfloor u \rfloor$ and $\lfloor u \rceil$ denotes u rounded up, down and to the closest integer.
- $|a + ib| = \sqrt{a^2 + b^2}$ where $a, b \in \mathbb{R}$ denotes the Euclidean norm of $a + ib$.
- $\|\mathbf{u}\|$ denotes the Euclidean norm of the vector $\mathbf{u} = (u_0, \dots, u_{n-1}) \in \mathbb{R}^n$.
- $u \sim v$ is used to denote that u and v are approximately of similar size.

1.4 Randomization

Given two group elements g and $x' = [d']g$ to be solved for d' , the general discrete logarithm problem may be randomized as follows:

1. Select a random integer t . Let $x = x' \odot [t]g = [d]g$.
2. Solve g and x for $d \equiv d' + t \pmod{r}$ and optionally for r .
3. Compute and return $d' \equiv d - t \pmod{r}$.

Hence, we may assume without loss of generality that d is selected uniformly at random on $[0, r)$ in the analysis of the quantum algorithm.

If r is known, t should be selected uniformly at random on $[0, r)$, otherwise on $[0, 2^{m+c})$ for m the bit length of r and c a sufficiently large integer constant for the selection of x to be indistinguishable from a uniform selection from $\mathbb{G}$. Solving for r in step 2 is only necessary if r is unknown and d' must be on $[0, r)$ when returned.

2 The quantum algorithm

In this section we describe the quantum algorithm, that upon input of a generator g and an element $x = [d]g$, where $0 \leq d < r$, outputs a pair (j, k) and element y .

As stated earlier, the algorithm is parameterized under a small integer constant $s \geq 1$, referred to as the tradeoff factor, that controls the tradeoff between the number of runs required and the requirements imposed on the quantum computer.

1. Let m be the integer such that $2^{m-1} \leq r < 2^m$, let $\ell = \lceil m/s \rceil$, and let

$$\Psi = \frac{1}{\sqrt{2^{m+2\ell}}} \sum_{a=0}^{2^{m+\ell}-1} \sum_{b=0}^{2^\ell-1} |a, b, 0\rangle.$$

2. Compute $[a]g \odot [-b]x = [a - bd]g$ to the third register to obtain

$$\Psi = \frac{1}{\sqrt{2^{m+2\ell}}} \sum_{a=0}^{2^{m+\ell}-1} \sum_{b=0}^{2^\ell-1} |a, b, [a - bd]g\rangle.$$

3. Compute QFTs of size $2^{m+\ell}$ and 2^ℓ of the first two registers to obtain

$$\Psi = \frac{1}{2^{m+2\ell}} \sum_{a=0}^{2^{m+\ell}-1} \sum_{b=0}^{2^\ell-1} \sum_{j=0}^{2^{m+\ell}-1} \sum_{k=0}^{2^\ell-1} e^{2\pi i (aj + 2^m bk)/2^{m+\ell}} |j, k, [a - bd]g\rangle.$$

4. Observe the system to obtain (j, k) and $y = [e]g$ where $e = (a - bd) \bmod r$.

Note that y is observed only to highlight that the system is forced to collapse to combinations of a and b such that $e = (a - bd) \bmod r$ for fixed e .

The above steps may be interleaved, rather than executed sequentially, so as to allow the qubits in the first two registers to be recycled [10, 18, 19]. A single control qubit then suffices to implement the first two control

registers. This is possible as the qubits in the control registers are not initially entangled; the registers are initialized to uniform superpositions of $2^{m+\ell}$ and 2^ℓ values, respectively.

In Shor's algorithm for computing general discrete logarithms, the two control registers are instead of length m qubits. Both registers are initialized to uniform superpositions of r values. This makes the single control qubit optimization less straightforward to apply, and the initial superpositions harder to induce. Apart from this difference, the implementation complexity of Shor's algorithm and our algorithm may be compared in a fair manner in terms of the total exponent lengths.

In practice, the exponentiation of group elements would typically be performed by computing a group operation controlled by each bit in the exponent. Hence, a total of $2m$ group operations are performed in Shor's algorithm, compared to $m + 2m/s$ in our algorithm. As s increases, this tends to m operations, providing an advantage over Shor's original algorithm by up to a factor of two at the expense of having to execute the algorithm multiple times. This reduction in the number of group operations translates into a corresponding reduction in the coherence time and circuit depth requirements of our quantum algorithm.

Note that our algorithm does not require r to be known. It suffices that the size of r is known. For comparison, Shor's algorithm requires r to be known. This explains why Shor needs to perform only $2m$ operations, whilst we need $3m$ operations when not making tradeoffs. As we shall see, we do in fact compute both d and r simultaneously, whilst Shor computes d given r .

3 The probability of observing (j, k) and y

In step 4 of the algorithm in Section 2, we obtain (j, k) and $y = [e]g$ with probability

$$\frac{1}{2^{2(m+\ell)}} \left| \sum_a \sum_b \exp \left[\frac{2\pi i}{2^{m+\ell}} (aj + 2^m bk) \right] \right|^2 \quad (1)$$

where the sum is over all pairs (a, b) , such that $0 \leq a < 2^{m+\ell}$ and $0 \leq b < 2^\ell$, respecting the condition $e \equiv a - bd \pmod{r}$. In this section, we seek a closed-form error-bounded approximation to (1) summed over all $y = [e]g \in \mathbb{G}$. To this end, we first perform a variable substitution to obtain contiguous summation intervals.

As $a = e + bd + n_r r$ for n_r an integer such that $0 \leq a = e + bd + n_r r < 2^{m+\ell}$, it follows that

$$\lceil -(e + bd)/r \rceil \leq n_r < \lceil (2^{m+\ell} - (e + bd))/r \rceil. \quad (2)$$

Substituting a for $e + bd + n_r r$ in (1) and adjusting the phase therefore yields

$$\frac{1}{2^{2(m+\ell)}} \left| \sum_{b=0}^{2^\ell-1} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} \exp \left[\frac{2\pi i}{2^{m+\ell}} (n_r r j + b(dj + 2^m k)) \right] \right|^2. \quad (3)$$

By introducing arguments α_d and α_r , and corresponding angles θ_d and θ_r , where

$$\alpha_d = \{dj + 2^m k\}_{2^{m+\ell}} \quad \alpha_r = \{rj\}_{2^{m+\ell}} \quad \theta_d = \theta(\alpha_d) = \frac{2\pi\alpha_d}{2^{m+\ell}} \quad \theta_r = \theta(\alpha_r) = \frac{2\pi\alpha_r}{2^{m+\ell}}$$

we may write (3) as a function of α_d and α_r , and e , as

$$\frac{1}{2^{2(m+\ell)}} \left| \sum_{b=0}^{2^\ell-1} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} \exp \left[\frac{2\pi i}{2^{m+\ell}} (n_r \alpha_r + b \alpha_d) \right] \right|^2 \quad (4)$$

or of θ_d and θ_r , and e , as

$$\rho(\theta_d, \theta_r, e) = \frac{1}{2^{2(m+\ell)}} \left| \sum_{b=0}^{2^\ell-1} e^{i\theta_d b} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} e^{i\theta_r n_r} \right|^2. \quad (5)$$

This implies that the probability of observing the pair (j, k) and $y = [e]g$ depends only on (α_d, α_r) and e , or equivalently on (θ_d, θ_r) and e . The probability is virtually independent of e in practice, as e can at most shift the endpoints of the summation interval in the inner sums in (4) and (5) by one step.

As was stated above, we seek a closed-form approximation to $\rho(\theta_d, \theta_r, e)$ summed over all r group elements $y = [e]g \in \mathbb{G}$. Hereinafter, we denote this probability

$$P(\theta_d, \theta_r) = \sum_{e=0}^{r-1} \rho(\theta_d, \theta_r, e) = \frac{1}{2^{2(m+2\ell)}} \sum_{e=0}^{r-1} \left| \sum_{b=0}^{2^\ell-1} e^{i\theta_d b} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} e^{i\theta_r n_r} \right|^2,$$

and we furthermore use angles and arguments interchangeably, depending on which representation best lends itself to analysis in each step of the process.

3.1 Preliminaries

To gain some intuition, we write $\rho(\theta_d, \theta_r, e)$ as

$$\frac{1}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^\ell-1} e^{i(\theta_d b + \theta_r \lceil -(e+bd)/r \rceil)} \sum_{n_r=0}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil - \lceil -(e+bd)/r \rceil - 1} e^{i\theta_r n_r} \right|^2$$

and note that there are two obstacles to placing this expression on closed form:

Firstly, the summation interval in the inner sum over n_r depends on the summation variable b of the outer sum. Secondly, the exponent of the summand in the outer sum over b contains a rounding operation that depends on b .

By using that $\lceil (2^{m+\ell} - (e + bd))/r \rceil - \lceil -(e + bd)/r \rceil \approx \lceil 2^{m+\ell}/r \rceil$ we may remove the dependency between the inner and outer sums, and by using that $\lceil -(e + bd)/r \rceil \approx -(e + bd)/r$ we may remove the rounding operation.

By making these two approximations, and by adjusting the phase, we may derive an approximation to $\rho(\theta_d, \theta_r, e)$ that is independent of e , enabling us to sum $\rho(\theta_d, \theta_r, e)$ over the r values of e , corresponding to the r group elements $y = [e]g \in \mathbb{G}$, simply by multiplying by r . This yields

$$\begin{aligned} P(\theta_d, \theta_r) &\approx \frac{r}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^\ell-1} e^{i(\theta_d - \theta_r d/r)b} \right|^2 \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right|^2 \\ &= \frac{r}{2^{2(m+2\ell)}} \left| \frac{e^{i2^\ell(\theta_d - \theta_r d/r)} - 1}{e^{i(\theta_d - \theta_r d/r)} - 1} \right|^2 \left| \frac{e^{i\lceil 2^{m+\ell}/r \rceil \theta_r} - 1}{e^{i\theta_r} - 1} \right|^2 \end{aligned} \quad (6)$$

where we furthermore need to assume in (6) that $\theta_d - \theta_r d/r \neq 0$ and $\theta_r \neq 0$.

This closed-form approximation captures the general characteristics of the probability distribution induced by the quantum algorithm. However, it is seemingly non-trivial to derive a good bound for the error in this approximation.

In what follows, we use techniques similar to those employed above to derive an error-bounded closed-form approximation to $\rho(\theta_d, \theta_r, e)$ such that the error is negligible in the regions of the plane where the probability mass is concentrated. As was the case above, we will find that the error-bounded approximation of $\rho(\theta_d, \theta_r, e)$ is independent of e , enabling us to approximate $P(\theta_d, \theta_r)$ simply by multiplying the closed-form approximation to $\rho(\theta_d, \theta_r, e)$ by r .

3.1.1 Constructive interference

Before we proceed to develop the closed-form approximation, we note that for a fixed problem instance and fixed e , the sums in $\rho(\theta_d, \theta_r, e)$ are over a constant number of unit vectors in the complex plane. For such sums, constructive interference arises when all vectors point in approximately the same direction.

In regions of the plane where θ_r and $\theta_d - \theta_r d/r$ are both small, we hence expect constructive interference to arise. The probability mass is expected to concentrate in regions where constructive interference arises, and where the concentration of pairs (θ_d, θ_r) yielded by the integers pairs (j, k) is great.

In what follows, we therefore seek to derive a closed-form approximation to $\rho(\theta_d, \theta_r, e)$, and an associated bound on the error in the approximation, such that the error is small when θ_r and $\theta_d - \theta_r d/r$ are small.

3.2 Closed-form approximation with error bounds

To derive a closed-form approximation to $\rho(\theta_d, \theta_r, e)$, we first observe that the sums in the expression for $\rho(\theta_d, \theta_r, e)$ may be regarded as sums over the points in a region $\mathcal{R}$ in a lattice $L^{a,b}$, as is illustrated in Figure 1. Note that this figure also contains other elements to which we shall return as the analysis progresses.

Figure 1: The lattice $L^{a,b}$ for $\sigma = 2$, $m = \ell = 5$, $e = 0$, $r = 31$ and $d = 27$. All red filled points are in $\mathcal{R}$. The region $\mathcal{A}$ and its translated replicas are drawn as dashed rectangles. All blue outlined points are in $\mathcal{A}$ or in one of its replicas. The gray triangles outline the points that are in $\mathcal{A}$ or one of its replicas, but not in $\mathcal{R}$, and vice versa.

Definition 3.1. Let $L^{a,b}$ be the lattice spanned by $(d, 1)$ and $(r, 0)$ so that the set of points in $L^{a,b}$ is given by $(a - e, b) = b(d, 1) + n_r(r, 0)$ for integers b and n_r .

Definition 3.2. Let $\mathcal{R}$ be the region in $L^{a,b}$ where $0 \leq a < 2^{m+\ell}$ and $0 \leq b < 2^\ell$.

Definition 3.3. Let

$$S_{\mathcal{R}} = \frac{|s_{\mathcal{R}}|^2}{2^{2(m+2\ell)}} \quad \text{where} \quad s_{\mathcal{R}} = \sum_{(a,b) \in \mathcal{R}} \exp \left[\frac{2\pi i}{2^{m+\ell}} (aj + 2^m bk) \right].$$

Claim 3.4. The probability $\rho(\theta_d, \theta_r, e) = S_{\mathcal{R}}$.

Proof. The points in $\mathcal{R}$ are given by $(a - e, b) = b(d, 1) + n_r(r, 0)$, for $0 \leq b < 2^\ell$ and n_r on (2) so that $0 \leq a = e + bd + n_r r < 2^{m+\ell}$, which implies that

$$\begin{aligned} S_{\mathcal{R}} &= \frac{1}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^\ell-1} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} \exp \left[\frac{2\pi i}{2^{m+\ell}} (n_r r j + b(dj + 2^m k)) \right] \right|^2 \\ &= \frac{1}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^\ell-1} e^{i\theta_d b} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} e^{i\theta_r n_r} \right|^2 = \rho(\theta_d, \theta_r, e) \end{aligned}$$

by the preliminary analysis in Section 3 and so the claim follows. $\square$

In what follows, we derive a closed-form approximation to $\rho(\theta_d, \theta_r, e) = S_{\mathcal{R}}$, and an associated error bound, in three steps.

3.2.1 Preliminaries

Before proceeding as outlined above, we first introduce some preliminary claims.

Claim 3.5. For $u, v \in \mathbb{C}$ and $\Delta = u - v$ it holds that

$$\left| |u|^2 - |v|^2 \right| \leq 2 |u| |\Delta| + |\Delta|^2.$$

Proof. First verify that

$$\begin{aligned} |u|^2 - |v|^2 &= |u|^2 - |u - \Delta|^2 = u\bar{u} - (u - \Delta)\overline{(u - \Delta)} \\ &= u\bar{u} - (u - \Delta)(\bar{u} - \bar{\Delta}) = u\bar{\Delta} + \bar{u}\Delta - |\Delta|^2 \end{aligned}$$

where the overlines denote complex conjugates. This implies that

$$\left| |u|^2 - |v|^2 \right| \leq |u| |\bar{\Delta}| + |\bar{u}| |\Delta| + |\Delta|^2 = 2 |u| |\Delta| + |\Delta|^2$$

and so the claim follows. $\square$

Claim 3.6. $|e^{i\phi} - 1| \leq |\phi|$ for any $\phi \in \mathbb{R}$.

Proof. It suffices to show that $|e^{i\phi} - 1|^2 = 2(1 - \cos \phi) \leq \phi^2$ from which the claim follows as $\cos \phi \geq 1 - \phi^2/2$ for any $\phi \in \mathbb{R}$. $\square$

3.2.2 Bounding $|s_{\mathcal{R}}|$

Before proceeding to the first approximation step, we furthermore bound $|s_{\mathcal{R}}|$ in this section, as this bound is needed in the following analysis.

Lemma 3.7. The sum $s_{\mathcal{R}}$ is bounded by $|s_{\mathcal{R}}| \leq 2^{2\ell+1}$.

Proof. By Claim 3.4 the sum

$$s_{\mathcal{R}} = \sum_{b=0}^{2^{\ell}-1} e^{i\theta_d b} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} e^{i\theta_r n_r}$$

where the outer sum over b is over 2^{ℓ} values and the inner sum over n_r is over at most $2^{\ell+1}$ values by Claim 3.8 below. As $s_{\mathcal{R}}$ is a sum of at most $2^{2\ell+1}$ complex unit vectors, it follows that $|s_{\mathcal{R}}| \leq 2^{2\ell+1}$, and so the lemma follows. $\square$

Claim 3.8. For $\Delta = \lceil (2^{m+\ell} - (e + bd))/r \rceil - \lceil -(e + bd)/r \rceil$, it holds that

$$\Delta = \lceil 2^{m+\ell}/r \rceil - t \leq 2^{\ell+1} \quad \text{for some } t \in \{0, 1\}.$$

Proof. For some $f_1, f_2 \in [0, 1)$, it holds that

$$\begin{aligned} \Delta &= \left\lceil \left\lceil 2^{m+\ell}/r \right\rceil - f_1 + \lceil -(e + bd)/r \rceil - f_2 \right\rceil - \lceil -(e + bd)/r \rceil \\ &= \left\lceil 2^{m+\ell}/r \right\rceil + \underbrace{\lceil -(e + bd)/r \rceil - \lceil -(e + bd)/r \rceil}_{=0} + \lceil -f_1 - f_2 \rceil \\ &= \left\lceil 2^{m+\ell}/r \right\rceil - \lfloor f_1 + f_2 \rfloor = \left\lceil 2^{m+\ell}/r \right\rceil - t \end{aligned}$$

where $t = \lfloor f_1 + f_2 \rfloor \in \{0, 1\}$ as $f_1 + f_2 \in [0, 2)$. Furthermore, recall that $r \geq 2^{m-1}$. Hence, it follows that $2^{m+\ell}/r \leq 2^{\ell+1}$, so $\Delta = \lceil 2^{m+\ell}/r \rceil - t \leq 2^{\ell+1}$, and so the claim follows. $\square$

3.2.3 Approximating $S_{\mathcal{R}}$ by $S_{\mathcal{A}} T_{\mathcal{A}}$

In the first approximation step, we approximate $S_{\mathcal{R}}$ by first summing the points in a small region $\mathcal{A}$ in $\mathcal{R}$, and by then replicating and translating the points in $\mathcal{A}$, and the associated sum over these points, so as to approximately cover $\mathcal{R}$, see Figure 1.

Definition 3.9. Let $\mathcal{A}$ be the region in $L^{a,b}$ where $0 \leq a < 2^{m+\ell}$ and $0 \leq b < 2^{\sigma}$ for σ an integer parameter selected on $0 < \sigma < \ell$.

Definition 3.10. Let

$$S_{\mathcal{A}} = \frac{|s_{\mathcal{A}}|^2}{2^{2(m+2\ell)}} \quad \text{where} \quad s_{\mathcal{A}} = \sum_{(a,b) \in \mathcal{A}} \exp \left[\frac{2\pi i}{2^{m+\ell}} (aj + 2^m bk) \right].$$

Claim 3.11.

$$S_{\mathcal{A}} = \frac{1}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^{\sigma}-1} e^{i\theta_d b} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} e^{i\theta_r n_r} \right|^2.$$

Proof. The points in $\mathcal{A}$ are given by $(a - e, b) = b(d, 1) + n_r(r, 0)$ for $0 \leq b < 2^{\sigma}$ and n_r on (2) so that $0 \leq a = e + bd + n_r r < 2^{m+\ell}$, which implies that

$$\begin{aligned} S_{\mathcal{A}} &= \frac{1}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^{\sigma}-1} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} \exp \left[\frac{2\pi i}{2^{m+\ell}} (n_r r j + b(dj + 2^m k)) \right] \right|^2 \\ &= \frac{1}{2^{2(m+2\ell)}} \left| \sum_{b=0}^{2^{\sigma}-1} e^{i\theta_d b} \sum_{n_r=\lceil -(e+bd)/r \rceil}^{\lceil (2^{m+\ell}-(e+bd))/r \rceil-1} e^{i\theta_r n_r} \right|^2 \end{aligned}$$

in analogy with the analysis in Section 3, but with b on $0 \leq b < 2^\sigma$ as opposed to $0 \leq b < 2^\ell$, and so the claim follows. $\square$

To replicate and translate the points in $\mathcal{A}$ so as to approximately cover $\mathcal{R}$, we furthermore introduce $t_{\mathcal{A}}$ and $T_{\mathcal{A}}$, as defined below:

Definition 3.12. Let

$$T_{\mathcal{A}} = |t_{\mathcal{A}}|^2 \quad \text{where} \quad t_{\mathcal{A}} = \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(\theta_d 2^\sigma + \theta_r \lceil -2^\sigma d/r \rceil) t}.$$

The error when approximating $S_{\mathcal{R}}$ by $S_{\mathcal{A}} T_{\mathcal{A}}$ may now be bounded as follows:

Lemma 3.13. *The error when approximating $s_{\mathcal{R}}$ by $s_{\mathcal{A}} t_{\mathcal{A}}$ is bounded by*

$$|s_{\mathcal{R}} - s_{\mathcal{A}} t_{\mathcal{A}}| \leq 2^{2\ell-\sigma+1}.$$

Proof. The exponential sum $t_{\mathcal{A}}$ replicates and translates the partial sum over $\mathcal{A}$ so as to approximately cover $\mathcal{R}$ as is illustrated in Figure 1. Every time the region is replicated, it is translated by a vector in $L^{a,b}$ that corresponds to $e^{i(\theta_d 2^\sigma + \theta_r \lceil -2^\sigma d/r \rceil)}$.

The error that arises when $s_{\mathcal{R}}$ is approximated by $s_{\mathcal{A}} t_{\mathcal{A}}$ is hence due to points that are in $\mathcal{R}$ but excluded from the sum, and conversely to points not in $\mathcal{R}$ that are erroneously included in the sum. Hereinafter these points will be referred to as the erroneous points.

The erroneous points fall within the two gray triangles in Figure 1. Both triangles are of horizontal side length 2^ℓ and vertical side length $2^{\ell-\sigma}(2^\sigma d \bmod r)$, as the region $\mathcal{A}$ is replicated and translated $2^{\ell-\sigma}$ times in total, and as it is shifted horizontally by 2^σ and vertically by $2^\sigma d \bmod r$ every time it is translated.

To upper-bound the number of lattice points in each triangle, note that the lattice points are on 2^ℓ vertical lines, evenly separated horizontally by a distance of one. The points on each vertical line are evenly separated vertically by a distance of r , with varying starting positions on each line. For $h(b) = 2^{\ell-\sigma}(2^\sigma d \bmod r)(b/2^\ell)$ the height of each triangle at b , we have that at most

$$N(b) = 1 + \left\lfloor \frac{h(b)}{r} \right\rfloor \leq 1 + \frac{h(b)}{r} = 1 + \frac{2^\sigma d \bmod r}{r} \frac{b}{2^\sigma} \leq 1 + \frac{b}{2^\sigma}$$

lattice points are then on the vertical line that cuts through the triangle at b , as may be seen by maximizing over all possible starting points. By summing $N(b)$ over all 2^ℓ lines, we thus obtain an upper bound of

$$\sum_{b=0}^{2^\ell-1} N(b) \leq 2^\ell + \frac{1}{2^\sigma} \sum_{b=0}^{2^\ell-1} b = 2^\ell + \frac{1}{2^\sigma} \frac{2^\ell(2^\ell-1)}{2} \leq 2^{2\ell-\sigma}$$

on the number of points in each triangle, where we have used that $2^{2\ell-\sigma-1} \geq 2^\ell$ as σ is an integer on $0 < \sigma < \ell$.

As there are two triangles, the total number of erroneous points is upper-bounded by $2 \cdot 2^{2\ell-\sigma} = 2^{2\ell-\sigma+1}$. Each erroneous point corresponds to a unit vector in the complex sum $s_{\mathcal{R}} - s_{\mathcal{A}} t_{\mathcal{A}}$, which implies $|s_{\mathcal{R}} - s_{\mathcal{A}} t_{\mathcal{A}}| \leq 2^{2\ell-\sigma+1}$, and so the lemma follows. $\square$

Lemma 3.14. *The error when approximating $S_{\mathcal{R}}$ by $S_{\mathcal{A}} T_{\mathcal{A}}$ is bounded by*

$$|S_{\mathcal{R}} - S_{\mathcal{A}} T_{\mathcal{A}}| \leq 2^{-2m-\sigma+4}.$$

Proof. By Claim 3.5, it holds that

$$\begin{aligned} \left| |s_{\mathcal{R}}|^2 - |s_{\mathcal{A}} t_{\mathcal{A}}|^2 \right| &\leq 2 |s_{\mathcal{R}}| |s_{\mathcal{R}} - s_{\mathcal{A}} t_{\mathcal{A}}| + |s_{\mathcal{R}} - s_{\mathcal{A}} t_{\mathcal{A}}|^2 \\ &\leq 2 \cdot 2^{2\ell+1} \cdot 2^{2\ell-\sigma+1} + 2^{4\ell-2\sigma+2} \leq 3 \cdot 2^{4\ell-\sigma+2} \leq 2^{4(\ell+1)-\sigma} \end{aligned}$$

as $|s_{\mathcal{R}} - s_{\mathcal{A}} t_{\mathcal{A}}| \leq 2^{2\ell-\sigma+1}$ by Lemma 3.13 and $|s_{\mathcal{R}}| \leq 2^{2\ell+1}$ by Lemma 3.7.

From the above, and Definitions 3.3, 3.10 and 3.12, we have that

$$|S_{\mathcal{R}} - S_{\mathcal{A}} T_{\mathcal{A}}| = \frac{||S_{\mathcal{R}}|^2 - |S_{\mathcal{A}} t_{\mathcal{A}}|^2|}{2^{2(m+2\ell)}} \leq \frac{2^{4(\ell+1)-\sigma}}{2^{2(m+2\ell)}} = 2^{-2m-\sigma+4}$$

and so the lemma follows. $\square$

As $t_{\mathcal{A}}$ is a geometric sum $T_{\mathcal{A}} = |t_{\mathcal{A}}|^2$ may be placed on closed form. It remains to derive a closed-form approximation to $S_{\mathcal{A}}$. In what follows, we do this in two additional approximation steps.

3.2.4 Approximating $S_{\mathcal{A}}$ by $S'_{\mathcal{A}}$

In the second approximation step, we derive a closed-form approximation to $S_{\mathcal{A}}$, by first approximating $S_{\mathcal{A}}$ by the product $S'_{\mathcal{A}}$ of two sums, such that the leading sum may be placed on closed form, and such that the trailing sum may be placed on closed form by means of a third approximation step.

Definition 3.15. Let

$$S'_{\mathcal{A}} = \frac{|S'_{\mathcal{A}}|^2}{2^{2(m+2\ell)}} \quad \text{where} \quad s'_{\mathcal{A}} = \sum_{b=0}^{2^{\sigma}-1} e^{i(\theta_d b + \theta_r \lceil -(e+bd)/r \rceil)} \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r}.$$

Lemma 3.16. The error when approximating $s_{\mathcal{A}}$ by $s'_{\mathcal{A}}$ is bounded by

$$|s_{\mathcal{A}} - s'_{\mathcal{A}}| \leq 2^{\sigma}.$$

Proof. As $s_{\mathcal{A}}$ and $s'_{\mathcal{A}}$ are sums of complex unit vectors, and as the sums differ by at most 2^{σ} vectors, as may be seen by comparing the summation intervals using Claim 3.8, it follows that $|s_{\mathcal{A}} - s'_{\mathcal{A}}| \leq 2^{\sigma}$, and so the lemma follows. $\square$

Lemma 3.17. The sum $s'_{\mathcal{A}}$ is bounded by $|s'_{\mathcal{A}}| \leq 2^{\ell+\sigma+1}$.

Proof. In the expression for $s'_{\mathcal{A}}$ in Definition 3.15, the sum over b assumes 2^{σ} values and the sum over n_r assumes at most $2^{\ell+1}$ values as the order $r \geq 2^{m-1}$. As $s'_{\mathcal{A}}$ is a sum of at most $2^{\ell+\sigma+1}$ complex unit vectors, it follows that $|s'_{\mathcal{A}}| \leq 2^{\ell+\sigma+1}$, and so the lemma follows. $\square$

Lemma 3.18. The error when approximating $S_{\mathcal{A}}$ by $S'_{\mathcal{A}}$ is bounded by

$$|S_{\mathcal{A}} - S'_{\mathcal{A}}| \leq 2^{-2m-3\ell+2\sigma+3}.$$

Proof. By Claim 3.5, it holds that

$$\begin{aligned} ||s_{\mathcal{A}}|^2 - |s'_{\mathcal{A}}|^2| &\leq 2 |s'_{\mathcal{A}}| |s_{\mathcal{A}} - s'_{\mathcal{A}}| + |s_{\mathcal{A}} - s'_{\mathcal{A}}|^2 \\ &\leq 2 \cdot 2^{\ell+\sigma+1} \cdot 2^{\sigma} + 2^{2\sigma} \leq 3 \cdot 2^{\ell+2\sigma+1} \leq 2^{\ell+2\sigma+3} \end{aligned}$$

as $|s_{\mathcal{A}} - s'_{\mathcal{A}}| \leq 2^{\sigma}$ by Lemma 3.16 and $|s'_{\mathcal{A}}| \leq 2^{\ell+\sigma+1}$ by Lemma 3.17.

From the above, and Definitions 3.10 and 3.15, we have that

$$|S_{\mathcal{A}} - S'_{\mathcal{A}}| = \frac{||s_{\mathcal{A}}|^2 - |s'_{\mathcal{A}}|^2|}{2^{2(m+2\ell)}} \leq \frac{2^{\ell+2\sigma+3}}{2^{2(m+2\ell)}} = 2^{-2m-3\ell+2\sigma+3}$$

and so the lemma follows. $\square$

The trailing sum in $s'_{\mathcal{A}}$ is geometric. Hence, it may be trivially placed on closed form. Due to the rounding operation in the exponent, this approach is not valid for the leading sum; we need a third approximation step.

3.2.5 Approximating $S'_{\mathcal{A}}$ by $S''_{\mathcal{A}}$

For θ_d and θ_r such that the angles $\theta_d b + \theta_r \lceil -(e + bd)/r \rceil \approx (\theta_d - \theta_r d/r) b$ in the leading sum in $S'_{\mathcal{A}}$ are small for all b on $0 \leq b < 2^\sigma$, all 2^σ terms in the sum are approximately one. In the third and final step of the approximation, we bound the error when simply approximating all terms in the leading sum by one.

Definition 3.19. Let

$$S''_{\mathcal{A}} = \frac{|S'_{\mathcal{A}}|^2}{2^{2(m+2\ell)}} \quad \text{where} \quad s''_{\mathcal{A}} = 2^\sigma \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r}.$$

Lemma 3.20. The difference between $s'_{\mathcal{A}}$ and $s''_{\mathcal{A}}$ is bounded by

$$|s'_{\mathcal{A}} - s''_{\mathcal{A}}| \leq 2^{\sigma-1} (|\theta_d| + |\theta_r|) |s''_{\mathcal{A}}|.$$

Proof. First observe that

$$|s'_{\mathcal{A}} - s''_{\mathcal{A}}| = \underbrace{\left| \sum_{b=0}^{2^\sigma-1} \left(e^{i(\theta_d b + \theta_r \lceil -(e+bd)/r \rceil)} - 1 \right) \right|}_{|\Delta|} \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right|.$$

By using Claim 3.6 and the triangle inequality, it follows that

$$\begin{aligned} |\Delta| &= \left| \sum_{b=0}^{2^\sigma-1} \left(e^{i(\theta_d b + \theta_r \lceil -(e+bd)/r \rceil)} - 1 \right) \right| \leq \sum_{b=0}^{2^\sigma-1} \left| e^{i(\theta_d b + \theta_r \lceil -(e+bd)/r \rceil)} - 1 \right| \\ &\leq \sum_{b=0}^{2^\sigma-1} |\theta_d b + \theta_r \lceil -(e+bd)/r \rceil| = \sum_{b=0}^{2^\sigma-1} |\theta_d b - \theta_r \lfloor (e+bd)/r \rfloor| \\ &\leq (|\theta_d| + |\theta_r|) \sum_{b=0}^{2^\sigma-1} b \leq (|\theta_d| + |\theta_r|) \frac{2^\sigma(2^\sigma-1)}{2} \leq 2^{2\sigma-1} (|\theta_d| + |\theta_r|) \end{aligned}$$

where we use that $\lceil -x \rceil = -\lfloor x \rfloor$ and $\lfloor (e+bd)/r \rfloor \leq b$. To verify the latter claim, note that $f_1 = e/r \in [0, 1)$ and $f_2 = bd/r \in [0, b)$ as $e, d \in [0, r)$. This implies that $\lfloor (e+bd)/r \rfloor = \lfloor f_1 + f_2 \rfloor \in [0, b]$ as $f_1 + f_2 \in [0, b+1)$.

By combining the above results, we now have that

$$|s'_{\mathcal{A}} - s''_{\mathcal{A}}| \leq 2^{2\sigma-1} (|\theta_d| + |\theta_r|) \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right| = 2^{\sigma-1} (|\theta_d| + |\theta_r|) |s''_{\mathcal{A}}|$$

and so the lemma follows. $\square$

Lemma 3.21. The error when approximating $S'_{\mathcal{A}}$ by $S''_{\mathcal{A}}$ is bounded by

$$|S'_{\mathcal{A}} - S''_{\mathcal{A}}| \leq 2^{\sigma-1} (|\theta_d| + |\theta_r|) \left(2 + 2^{\sigma-1} (|\theta_d| + |\theta_r|) \right) S''_{\mathcal{A}}.$$

Proof. By Claim 3.5, it holds that

$$\begin{aligned} \left| |s'_{\mathcal{A}}|^2 - |s''_{\mathcal{A}}|^2 \right| &\leq 2 |s''_{\mathcal{A}}| |s'_{\mathcal{A}} - s''_{\mathcal{A}}| + |s'_{\mathcal{A}} - s''_{\mathcal{A}}|^2 \\ &\leq 2 \cdot 2^{\sigma-1} (|\theta_d| + |\theta_r|) |s''_{\mathcal{A}}|^2 + 2^{2(\sigma-1)} (|\theta_d| + |\theta_r|)^2 |s''_{\mathcal{A}}|^2 \\ &= 2^{\sigma-1} (|\theta_d| + |\theta_r|) \left(2 + 2^{\sigma-1} (|\theta_d| + |\theta_r|) \right) |s''_{\mathcal{A}}|^2 \end{aligned}$$

as $|s'_{\mathcal{A}} - s''_{\mathcal{A}}| \leq 2^{\sigma-1} (|\theta_d| + |\theta_r|) |s''_{\mathcal{A}}|$ by Lemma 3.20.

From the above, and Definitions 3.15 and 3.19, we have that

$$|S'_{\mathcal{A}} - S''_{\mathcal{A}}| = \frac{|S'_{\mathcal{A}}|^2 - |S''_{\mathcal{A}}|^2}{2^{2(m+2\ell)}} \leq 2^{\sigma-1} (|\theta_d| + |\theta_r|) \left(2 + 2^{\sigma-1} (|\theta_d| + |\theta_r|)\right) S''_{\mathcal{A}}$$

and so the lemma follows. $\square$

This yields an approximation $S''_{\mathcal{A}}$ to $S'_{\mathcal{A}}$ that may be placed on closed form.

3.2.6 Main approximability result

By combining the previous results, the main approximability result follows:

Theorem 3.22. *The probability $P(\theta_d, \theta_r)$ of observing a specific pair (j, k) with angle pair (θ_d, θ_r) , summed over all $y \in \mathbb{G}$, may be approximated by*

$$\begin{aligned} \tilde{P}(\theta_d, \theta_r) &= \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} \left| \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(\theta_d 2^{\sigma} + \theta_r \lceil -2^{\sigma} d/r \rceil) t} \right|^2 \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right|^2 \\ &= \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} \left| \frac{e^{i(\theta_d 2^{\sigma} + \theta_r \lceil -2^{\sigma} d/r \rceil) 2^{\ell-\sigma}} - 1}{e^{i(\theta_d 2^{\sigma} + \theta_r \lceil -2^{\sigma} d/r \rceil)} - 1} \right|^2 \left| \frac{e^{i\theta_r \lceil 2^{m+\ell}/r \rceil} - 1}{e^{i\theta_r} - 1} \right|^2 \end{aligned}$$

assuming $\theta_d 2^{\sigma} + \theta_r \lceil -2^{\sigma} d/r \rceil \neq 0$ and $\theta_r \neq 0$ when placing the expression on closed form. The approximation error $|P(\theta_d, \theta_r) - \tilde{P}(\theta_d, \theta_r)| \leq \tilde{e}(\theta_d, \theta_r)$ where

$$\tilde{e}(\theta_d, \theta_r) = \frac{2^4}{2^{m+\sigma}} + \frac{2^3}{2^{m+\ell}} + \frac{2^{\sigma}}{2} (|\theta_d| + |\theta_r|) \left(2 + \frac{2^{\sigma}}{2} (|\theta_d| + |\theta_r|)\right) \tilde{P}(\theta_d, \theta_r).$$

Proof. The probability $\rho(\theta_d, \theta_r, e)$ of observing a specific pair (j, k) , with angle pair (θ_d, θ_r) , and some group element $y = [e] g \in \mathbb{G}$, is $S_{\mathcal{R}}$ by Claim 3.4.

The error when approximating $S_{\mathcal{R}}$ by $S_{\mathcal{A}} T_{\mathcal{A}}$ is bounded by

$$|S_{\mathcal{R}} - S_{\mathcal{A}} T_{\mathcal{A}}| \leq 2^{-2m-\sigma+4}$$

by Lemma 3.14. The error when approximating $S_{\mathcal{A}} T_{\mathcal{A}}$ by $S'_{\mathcal{A}} T_{\mathcal{A}}$ is bounded by

$$|S_{\mathcal{A}} T_{\mathcal{A}} - S'_{\mathcal{A}} T_{\mathcal{A}}| \leq 2^{-2m-3\ell+2\sigma+3} T_{\mathcal{A}}$$

by Lemma 3.18. The error when approximating $S'_{\mathcal{A}} T_{\mathcal{A}}$ by $S''_{\mathcal{A}} T_{\mathcal{A}}$ is bounded by

$$|S'_{\mathcal{A}} T_{\mathcal{A}} - S''_{\mathcal{A}} T_{\mathcal{A}}| \leq 2^{\sigma-1} (|\theta_d| + |\theta_r|) (2 + 2^{\sigma-1} (|\theta_d| + |\theta_r|)) S''_{\mathcal{A}} T_{\mathcal{A}}$$

by Lemma 3.21. By the triangle inequality

$$\begin{aligned} |S_{\mathcal{R}} - S''_{\mathcal{A}} T_{\mathcal{A}}| &= |(S_{\mathcal{R}} - S_{\mathcal{A}} T_{\mathcal{A}}) + (S_{\mathcal{A}} T_{\mathcal{A}} - S'_{\mathcal{A}} T_{\mathcal{A}}) + (S'_{\mathcal{A}} T_{\mathcal{A}} - S''_{\mathcal{A}} T_{\mathcal{A}})| \\ &\leq |S_{\mathcal{R}} - S_{\mathcal{A}} T_{\mathcal{A}}| + T_{\mathcal{A}} |S_{\mathcal{A}} - S'_{\mathcal{A}}| + T_{\mathcal{A}} |S'_{\mathcal{A}} - S''_{\mathcal{A}}|. \end{aligned}$$

Neither of these three error terms, nor the expression for $S''_{\mathcal{A}} T_{\mathcal{A}}$, depend on e . Hence, we may sum over all r elements $y = [e] g \in \mathbb{G}$ by multiplying by r . It therefore follows that $\tilde{P}(\theta_d, \theta_r) = r S''_{\mathcal{A}} T_{\mathcal{A}}$ is an approximation to $P(\theta_d, \theta_r)$, and that the error that arises in this approximation is bounded by

$$\begin{aligned} r |S_{\mathcal{R}} - S_{\mathcal{A}} T_{\mathcal{A}}| + r T_{\mathcal{A}} |S_{\mathcal{A}} - S'_{\mathcal{A}}| + r T_{\mathcal{A}} |S'_{\mathcal{A}} - S''_{\mathcal{A}}| \\ \leq 2^{-2m-\sigma+4} r + 2^{-2m-3\ell+2\sigma+3} r T_{\mathcal{A}} + \\ 2^{\sigma-1} (|\theta_d| + |\theta_r|) (2 + 2^{\sigma-1} (|\theta_d| + |\theta_r|)) r S''_{\mathcal{A}} T_{\mathcal{A}} \end{aligned}$$

$$\leq \frac{2^4}{2^{m+\sigma}} + \frac{2^3}{2^{m+\ell}} + \frac{2^\sigma}{2}(|\theta_d| + |\theta_r|) \left(2 + \frac{2^\sigma}{2}(|\theta_d| + |\theta_r|)\right) \tilde{P}(\theta_d, \theta_r)$$

where we use that $r < 2^m$, and that $T_{\mathcal{A}} \leq 2^{2(\ell-\sigma)}$ as it is the square norm of a sum of $2^{\ell-\sigma}$ unit vectors by Definition 3.12, and so the theorem follows. $\square$

In Appendix C we demonstrate the soundness of this approximation.

4 The distribution of pairs (α_d, α_r)

In this section, we identify and count all pairs (j, k) that yield (α_d, α_r) and analyze the distribution and density of pairs (α_d, α_r) in the plane.

Definition 4.1. An argument pair (α_d, α_r) is said to be admissible if there exists an integer pair (j, k) , for j on $0 \leq j < 2^{m+\ell}$ and k on $0 \leq k < 2^\ell$, such that

$$\alpha_d = \{dj + 2^m k\}_{2^{m+\ell}} \quad \text{and} \quad \alpha_r = \{rj\}_{2^{m+\ell}}.$$

Definition 4.2. Let κ_d denote the greatest integer such that 2^{κ_d} divides d , and let κ_r denote the greatest integer such that 2^{κ_r} divides r .

Definition 4.3. Let L^α be the lattice generated by the rows in

$$\begin{bmatrix} \delta_r & 2^{\kappa_r} \\ 2^{m-\gamma} & 0 \end{bmatrix} \quad \text{where} \quad \delta_r = d \left(\frac{r}{2^{\kappa_r}} \right)^{-1} \pmod{2^{m-\gamma}}$$

and $\gamma = \max(0, \kappa_r - (\ell + \kappa_d))$.

Lemma 4.4. The admissible argument pairs (α_d, α_r) are vectors in L^α in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. There are $2^{m+2\ell-\kappa_r+\gamma}$ distinct admissible argument pairs, each occurring with multiplicity $2^{\kappa_r-\gamma}$.

Proof. As $\alpha_r \equiv rj \pmod{2^{m+\ell}}$, the set of integers j that yield α_r are given by

$$j \equiv \frac{\alpha_r}{2^{\kappa_r}} \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + 2^{m+\ell-\kappa_r} t_r \pmod{2^{m+\ell}}$$

as t_r runs through all integers on $0 \leq t_r < 2^{\kappa_r}$. As $\alpha_d \equiv dj + 2^m k \pmod{2^{m+\ell}}$, we need

$$\begin{aligned} \alpha_d &\equiv d \left(\frac{\alpha_r}{2^{\kappa_r}} \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + 2^{m+\ell-\kappa_r} t_r \right) + 2^m k \\ &\equiv \frac{\alpha_r}{2^{\kappa_r}} d \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + \underbrace{2^{m+\ell-\kappa_r+\kappa_d} \frac{dt_r}{2^{\kappa_d}}}_A + \underbrace{2^m k}_B \pmod{2^{m+\ell}} \end{aligned} \quad (7)$$

for k an integer on $0 \leq k < 2^\ell$, to ensure compatibility. As $2^{m-\gamma}$ is the largest power of two to divide both 2^m and $2^{m+\ell-\kappa_r+\kappa_d}$, by the definition of γ , the congruence relation $\alpha_d \equiv (\alpha_r/2^{\kappa_r}) d (r/2^{\kappa_r})^{-1} \pmod{2^{m-\gamma}}$ must hold.

As t_r and k run through all pairwise combinations, the set of $2^{\ell+\kappa_r}$ arguments α_d generated by (7) is equal to that generated by

$$\alpha_d \equiv \frac{\alpha_r}{2^{\kappa_r}} d \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + 2^{m-\gamma} t_\gamma \quad (8)$$

$$\equiv \frac{\alpha_r}{2^{\kappa_r}} \left(d \left(\frac{r}{2^{\kappa_r}} \right)^{-1} \pmod{2^{m-\gamma}} \right) + 2^{m-\gamma} t'_\gamma \pmod{2^{m+\ell}} \quad (9)$$

as t_γ , or equivalently t'_γ , runs through all integers on $0 \leq t_\gamma, t'_\gamma < 2^{\ell+\kappa_r}$.

To go from (7) to (8), first note that B runs through all values in $[2^m, 2^{m+\ell})$. If $\gamma = 0$, term A introduces multiplicity by repeating the sequence generated by B with various offsets. These offsets are of no significance to this analysis, as we only account for which values occur in the set and with what multiplicity.

If $\gamma > 0$, term A runs through all values in $[2^{m-\gamma}, 2^{m-\gamma+\kappa_r})$. As $\kappa_r \geq \gamma$ when $\gamma > 0$, term A runs through all values in the subrange $[2^{m-\gamma}, 2^m)$. When A assumes values greater than or equal to 2^m , it introduces multiplicity by repeating the sequence of all values on $[2^{m-\gamma}, 2^{m+\ell})$ generated by A and B with various offsets.

This implies that $(A + B) \bmod 2^{m+\ell}$ runs through all $2^{m+\ell}/2^{m-\gamma} = 2^{\ell+\gamma}$ values on $[2^{m-\gamma}, 2^{m+\ell})$ with multiplicity $2^{\ell+\kappa_r}/2^{\ell+\gamma} = 2^{\kappa_r-\gamma}$, and this is exactly what is stated in (8). To go from (8) to (9) is trivial.

As there are $2^{m+2\ell}$ admissible argument pairs, and as each pair occurs with multiplicity $2^{\kappa_r-\gamma}$, there are $2^{m+2\ell-\kappa_r+\gamma}$ distinct admissible argument pairs.

The lattice L^α is constructed from (9), as the admissible arguments α_r are multiples of 2^{κ_r} , and as the admissible $\alpha_d \equiv (\alpha_r / 2^{\kappa_r}) \delta_r + 2^{m+\gamma} t'_\gamma \pmod{2^{m+\ell}}$, in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$, and so the lemma follows. $\square$

In Figure 2 the distribution of admissible argument pairs (α_d, α_r) in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$ is depicted for various combinations of d and r .

Figure 2: The distribution of admissible argument pairs (α_d, α_r) in the region where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$ for $m = 4$ and $\ell = 3$, and example combinations of d and r , as indicated. The lattice may be constructed by replicating the fundamental parallelogram (blue) or a rectangle (gray) of size $2^{m-\gamma} \times 2^{m+\ell-\gamma}$.

4.1 Pairs (j, k) yielding (α_d, α_r)

In this section we identify all pairs (j, k) that yield (α_d, α_r) .

Lemma 4.5. *The set of integer pairs (j, k) , for j on $0 \leq j < 2^{m+\ell}$ and k on $0 \leq k < 2^\ell$, that yield the admissible argument pair (α_d, α_r) is given by*

$$j = \left(\frac{\alpha_r}{2^{\kappa_r}} \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + 2^{m+\ell-\kappa_r} t_r \right) \bmod 2^{m+\ell} \quad \text{and} \quad k = \frac{\alpha_d - dj}{2^m} \bmod 2^\ell$$

as t_r runs through all integer multiples of 2^γ on $0 \leq t_r < 2^{\kappa_r}$.

Proof. As $\alpha_r \equiv rj \pmod{2^{m+\ell}}$, solving for j yields

$$j = \left(\frac{\alpha_r}{2^{\kappa_r}} \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + 2^{m+\ell-\kappa_r} t_r \right) \bmod 2^{m+\ell}$$

for t_r an integer $0 \leq t_r < 2^{\kappa_r}$.

As $\alpha_d \equiv dj + 2^m k \pmod{2^{m+\ell}}$, for compatibility 2^m must divide $2^{m+\ell-\kappa_r} dt_r$ for all $t_r \neq 0$. As $2^{m+\ell+\kappa_d-\kappa_r}$ is the greatest power of two to divide $2^{m+\ell-\kappa_r} d$, it follows that t_r must be a multiple of 2^γ , and so the lemma follows. $\square$

4.2 The density of pairs (α_d, α_r)

In this section we analyze the density of admissible argument pairs (α_d, α_r) in the argument plane.

Claim 4.6. *The density of admissible argument pairs (α_d, α_r) in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$ is 2^{-m} when accounting for multiplicity.*

Proof. There are $2^{m+2\ell}$ admissible argument pairs (α_d, α_r) , when accounting for multiplicity, in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. This region is of area $2^{2(m+\ell)}$. The density is hence $2^{m+2\ell} / 2^{2(m+\ell)} = 2^{-m}$, and so the claim follows. $\square$

To construct the histogram for the probability distribution, the argument plane is divided into small rectangular subregions. Lemma 4.7 below bounds the error when approximating the density in such subregions by 2^{-m} .

Lemma 4.7. *Let D be the density of admissible argument pairs (α_d, α_r) , when accounting for multiplicity, in a rectangle R of area A and circumference C in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. Then*

$$\left| D - \frac{1}{2^m} \right| \leq 2^{\kappa_r-\gamma} \frac{2C\lambda_2 + 4(2\lambda_2)^2}{A \det L^\alpha} = \frac{2C\lambda_2 + 4(2\lambda_2)^2}{2^m A}$$

for λ_1 the norm of the shortest non-zero vector $\mathbf{w}_1 \in L^\alpha$, and λ_2 the norm of the shortest non-zero vector $\mathbf{w}_2 \in L^\alpha$ that is linearly independent to $\mathbf{w}_1$.

Proof. By Lemma 4.4, the admissible argument pairs (α_d, α_r) are vectors in L^α in the region of the argument plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. Each admissible argument pair occurs with multiplicity $2^{\kappa_r-\gamma}$.

The fundamental parallelogram in L^α contains a single lattice vector. It is spanned by $\mathbf{w}_1$ and $\mathbf{w}_2$, and has area $\det L^\alpha = \lambda_2 |\mathbf{w}_\perp| = 2^{m+\kappa_r-\gamma}$, where $\mathbf{w}_\perp$ is the component in $\mathbf{w}_1$ perpendicular to $\mathbf{w}_2$. This implies $\lambda_2 \geq \lambda_1 \geq |\mathbf{w}_\perp|$. To bound the number of argument pairs $(\alpha_d, \alpha_r) \in R$, we lower- and upper-bound the number of fundamental parallelograms that can at most fit into R , as described below, paying particular attention to the border areas:

To upper-bound the number of vectors in R , we extend each side of R by $2\lambda_2$ length units, to ensure that any parallelogram that is only partly in R is included in the count, and divide the area of the resulting rectangle by the area of the fundamental parallelogram. This yields $(A + 2C\lambda_2 + 4(2\lambda_2)^2) / \det L^\alpha$.

Conversely, to lower-bound the number of vectors in R , we retract each side of R by $2\lambda_2$ length units, to ensure that all parallelograms that are only partly in the rectangle are excluded from the count, and divide the area of the resulting rectangle by $\det L^\alpha$. This yields $(A - 2C\lambda_2 + 4(2\lambda_2)^2) / \det L^\alpha$.

By combining the upper and lower bounds, dividing by the area A of R , and multiplying by $2^{\kappa_r - \gamma}$ to account for multiplicity, the lemma follows. $\square$

For known d and r , Lemma 4.7 above provides a bound on the error when approximating the density in a rectangle in L^α by 2^{-m} as λ_2 may then be computed. To bound the error for general problem instances, and when d and r are unknown, we introduce the following less tight lemma:

Lemma 4.8. *Let D be the density of admissible argument pairs (α_d, α_r) , when accounting for multiplicity, in a rectangle of side lengths l_d and l_r in the α_d and α_r directions, respectively, in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. Then*

$$\left| D - \frac{1}{2^m} \right| \leq \frac{2^{\kappa_r}}{2^m l_r} + \frac{1}{2^\gamma l_d} + \frac{1}{l_d l_r}.$$

Proof. By Lemma 4.4, the admissible argument pairs (α_d, α_r) are vectors in L^α in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. Each admissible argument pair occurs with multiplicity $2^{\kappa_r - \gamma}$.

The vectors in L^α are on horizontal lines (for fixed α_r) evenly separated by a vertical distance of 2^{κ_r} . The number of such lines that intersect the rectangle is upper-bounded by $\lfloor l_r / 2^{\kappa_r} \rfloor + 1 \leq l_r / 2^{\kappa_r} + 1$ and lower-bounded by $\lfloor l_r / 2^{\kappa_r} \rfloor \geq l_r / 2^{\kappa_r} - 1$ as may be seen by positioning the rectangle to maximize or minimize the number of lines that intersect the rectangle.

On each line, the vectors in L^α are evenly spaced by a distance of $2^{m-\gamma}$ with varying starting positions. The number of vectors in L^α that fall within the rectangle on each line is upper-bounded by $\lfloor l_d / 2^{m-\gamma} \rfloor + 1 \leq l_d / 2^{m-\gamma} + 1$ and lower-bounded by $\lfloor l_d / 2^{m-\gamma} \rfloor \geq l_d / 2^{m-\gamma} - 1$, when not accounting for multiplicity, as may be seen by positioning the line to maximize or minimize the number of vectors that fall within the rectangle.

Hence the number of lattice vectors in the rectangle is upper-bounded by

$$2^{\kappa_r - \gamma} (l_r / 2^{\kappa_r} + 1) (l_d / 2^{m-\gamma} + 1) = l_d l_r / 2^m + l_d 2^{\kappa_r} / 2^m + l_r / 2^\gamma + 1$$

and lower-bounded by

$$2^{\kappa_r - \gamma} (l_r / 2^{\kappa_r} - 1) (l_d / 2^{m-\gamma} - 1) = l_d l_r / 2^m - l_d 2^{\kappa_r} / 2^m - l_r / 2^\gamma + 1$$

as each vector corresponds to a pair that occurs with multiplicity $2^{\kappa_r - \gamma}$. By combining these bounds, and dividing by the rectangle area $l_d l_r$, the lemma follows. $\square$

For unknown d and r , Lemma 4.8 above provides an error bound, assuming only some bounds on the parameters κ_r and γ . Asymptotically, the error in the approximation tends to zero as the side lengths of the rectangle tend to infinity. For rectangular subregions of specific dimensions, it may furthermore be shown that the error is zero, as is demonstrated in the following lemma:

Lemma 4.9. *The density of admissible argument pairs (α_d, α_r) in a rectangle of side lengths positive integer multiples of $2^{m-\gamma}$ and $2^{m-\gamma+\kappa_r}$ in α_d and α_r , respectively, in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$, is 2^{-m} when accounting for multiplicity.*

Proof. By Lemma 4.4, the admissible argument pairs (α_d, α_r) are vectors in L^α in the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$. Each admissible argument pair occurs with multiplicity $2^{\kappa_r - \gamma}$.

From the definition of L^α in Lemma 4.4, it follows that the lattice is cyclic with period $2^{m-\gamma}$ in α_d and $2^{m-\gamma+\kappa_r}$ in α_r . This is illustrated in Figure 2 where rectangular regions of these dimensions are highlighted in gray. The highlighted regions all extend from the origin in Figure 2 but the starting point may of course be arbitrarily selected. This implies that the lattice L^α may be generated by replicating and translating any rectangle of side lengths positive multiples of $2^{m-\gamma}$ and $2^{m-\gamma+\kappa_r}$ in α_d and α_r , respectively, see Figure 2,

throughout the plane. The same holds if the rectangle is replicated and translated cyclically throughout the region of the plane where $-2^{m+\ell-1} \leq \alpha_d, \alpha_r < 2^{m+\ell-1}$.

The number of rectangles that fit in the region when replicated and translated cyclically is

$$2^{2(m+\ell)} / 2^{2(m-\gamma)+\kappa_r} = 2^{2(\ell+\gamma)-\kappa_r}$$

as the area of the region is $2^{2(m+\ell)}$ and the area of the rectangle is $2^{2(m-\gamma)+\kappa_r}$. The total number of lattice vectors in the region is $2^{2m+\ell}$, so each rectangle contains $2^{m+2\ell} / 2^{2(\ell+\gamma)-\kappa_r} = 2^{m-2\gamma+\kappa_r}$ vectors when accounting for multiplicity. By dividing by the area of the rectangle, we see that the density of points in each rectangle is $2^{m-2\gamma+\kappa_r} / 2^{2(m-\gamma)+\kappa_r} = 2^{-m}$, and so the lemma follows. $\square$

5 Simulating the quantum algorithm

In close analogy with [7], we now proceed to construct a high-resolution histogram for the probability distribution induced by the quantum algorithm, for given d and r , and to sample it to simulate the quantum algorithm.

5.1 Constructing the histogram

Except for the fact that the probability distribution is two-dimensional, and that we need to account for the closed-form expression being an approximation, we exactly follow [7] to construct the high-resolution histogram: We subdivide the argument plane into regions and subregions, and integrate the closed-form probability approximation $\tilde{P}(\theta_d, \theta_r)$ and associated error bound $\tilde{e}(\theta_d, \theta_r)$ numerically in each subregion.

First, we subdivide each quadrant of the argument plane into $(30 + \mu)$ rectangular regions where $\mu = \min(\ell - 2, 11)$. Each region thus formed is uniquely identified by $(\eta_d, \eta_r) \in \mathbb{Z}^2$ by requiring that for all (α_d, α_r) in the region

$$2^{|\eta_d|} \leq |\alpha_d| \leq 2^{|\eta_d|+1} \quad \text{and} \quad 2^{|\eta_r|} \leq |\alpha_r| \leq 2^{|\eta_r|+1},$$

and furthermore that $\text{sgn}(\alpha_d) = \text{sgn}(\eta_d)$ and $\text{sgn}(\alpha_r) = \text{sgn}(\eta_r)$, where η_d and η_r are such that

$$m - 30 \leq |\eta_d|, |\eta_r| \leq m + \mu - 1,$$

see the illustration in Figure 3.

Figure 3: The subdivision of the plane into regions and subregions. The gray box illustrates Simpson's rule applied to a subregion. The probability is computed in the blue corner points, the four red border mid-points and the red center-point.

Then, we subdivide each region into rectangular subregions identified by an integer pair (ξ_d, ξ_r) by requiring that for all (α_d, α_r) in the subregion

$$2^{|\eta_d| + \xi_d/2^v} \leq |\alpha_d| \leq 2^{|\eta_d| + (\xi_d+1)/2^v} \quad \text{and} \quad 2^{|\eta_r| + \xi_r/2^v} \leq |\alpha_r| < 2^{|\eta_r| + (\xi_r+1)/2^v}$$

where $0 \leq \xi_d, \xi_r < 2^v$ for $v \in \{6, 7, 8, 9\}$ a resolution parameter adaptively selected as a function of the probability mass and variance in each region.

For each subregion, we compute the approximate probability mass contained within the subregion, and an associated error bound, by applying Simpson's rule in two dimensions, followed by Richardson extrapolation to cancel the linear error term, and division by 2^m to account for the density of pairs.

Simpson's rule is hence applied $2^{2v}(1 + 2^2)$ times in each region. Each application requires $\tilde{P}(\theta_d, \theta_r)$ and $\tilde{e}(\theta_d, \theta_r)$ to be evaluated in up to nine points, for which purpose we use the closed-form expressions in Theorem 3.22, with σ adaptively selected to suppress $\tilde{e}(\theta_d, \theta_r)$.

The optimal σ may be found by searching exhaustively. A computationally more efficient method for selecting σ is to use the heuristic in Appendix C.5.3. We use the heuristic in all cases except when s is large in relation to m causing the error in the closed-form approximation to be large. For such m and s we accept an extra computational burden to get slightly better σ and slightly smaller errors.

In what follows, we refer to the total probability mass captured as the sum of the integral of $\tilde{P}(\theta_d, \theta_r)$ over all subregions, and to the total approximation error as the sum of the integral of $\tilde{e}(\theta_d, \theta_r)$ over all subregions. Note that the total approximation error is an upper bound, by this definition, that is by no means tight. The actual error in the approximation is likely smaller than the bound indicates.

In order to save space when storing the histogram, we discard regions that capture insignificant shares of the probability mass. Note furthermore that for m and s such that the total approximation error is large, the error may often be reduced at the expense of capturing a smaller fraction of the probability mass by simply discarding selected regions where the error is large. The errors we report in this paper are without accounting for such additional filtering.

Note that this method of constructing the histogram assumes κ_d and κ_r to be small in relation to m . Note also that it follows from Section 4.2 that it is sound to approximate the density by 2^{-m} in the four regions of interest in the plane. For the m and s that we consider, the error in the density approximation is negligible.

5.2 Understanding the probability distribution

To illustrate the distribution that arises, a histogram is plotted in the signed logarithmic argument plane in Figure 4 for $m = 2048$ and $s = 30$, and for d and r selected as explained in Section 7.3. It captures approximately 99.99% of the probability mass. The total approximation error is less than 10^{-3} .

The histogram plotted in Figure 4 captures the characteristic appearance of the probability distribution when d and r are both of length m bits and not divisible by large powers of two.

The probability mass is located in the regions where $(|\alpha_d|, |\alpha_r|) \sim (2^m, 2^m)$, whereas for random pairs (j, k) the arguments would both be of size $\sim 2^{m+\ell}$. This implies that a single run of the quantum algorithm yields $\sim \ell \sim m/s$ bits of information on d and r , respectively.

The distribution is symmetric, in that the top right and lower left quadrants are mirrored, as are the top left and lower right quadrants. It hence suffices to compute only two quadrants to construct the histogram. To see why this is, note that flipping the sign of both arguments in the expression for $\tilde{P}(\theta_d, \theta_r)$ in Theorem 3.22 has no effect. Flipping the sign of only one argument, on the other hand, may lead to cancellation or lack of cancellation in the angle $\theta_d 2^\sigma + \theta_r \lceil -2^\sigma d/r \rceil$. This explains the concentration of probability mass in the top right and lower left quadrants, and in the tail that extends along the diagonal in Figure 4 where $\theta_d 2^\sigma + \theta_r \lceil -2^\sigma d/r \rceil$ is small.

If the size of d in relation to r is gradually reduced, the tail along the diagonal gradually disappears, as the cancellation effect weakens when d/r is reduced in size. When d is approximately of length $m - 5$ bits, the tail is nearly gone, and all four quadrants are nearly mirrored. Further reducing d has no significant effect on the distribution. Varying r on the interval $2^{m-1} < r < 2^m$ only slightly affects the distribution. Scaling m

Figure 4: The probability distribution for general discrete logarithms computed as in Section 5.1 for $m = 2048$, $s = 30$, and d and r selected as in Section 7.3. To facilitate printing, the resolution has been reduced in this figure.

Figure 5: The probability distribution for short discrete logarithms computed as in Appendix B from the closed-form expression in [7], for $m = 2048$ and $s = 30$, and d selected as in Section 7.3. The resolution has been reduced in this figure.

and s has virtually no effect on the distribution, for as long as m/s does not become so small so as to cause constructive interference not to arise, or the plot to be cropped.

The marginal distribution along the α_r axis in Figure 4 is virtually identical to the distribution induced by r when performing order finding, see Appendix A and Figure A1 for comparison. In Appendix D we prove this correspondence analytically by summing $\tilde{P}(\theta_d, \theta_r)$ over all admissible angles θ_d with multiplicity. Analogously, the marginal distribution along the α_d axis is seemingly virtually identical to the probability distribution induced by d when regarded as a short discrete logarithm, see [7] and Figure 5 for comparison. We have not as of yet been able to prove this correspondence analytically but it may be evidenced numerically by comparing the distributions for specific problem instances.

The above observations imply that the lattice-based post-processing algorithm introduced in [7] may be used to solve sets of pairs (j, k) for both short and general d , with minor modifications, see Section 6.1. An analogous lattice-based algorithm may be used to solve sets of integers j for r , see Section 6.2. The hardest instances to solve are those where d is large in relation to r , and r is large in relation to 2^m , as in Figure 4, due to the tail that then extends along the diagonal.

5.3 Sampling the probability distribution

Except for the fact that the probability distribution is two-dimensional, we exactly follow [7] to sample the distribution: To sample an argument pair (α_d, α_r) , we first sample a subregion and then sample (α_d, α_r) from this subregion.

To sample the subregion, we first order all subregions in the histogram by probability, and compute the cumulative probability up to and including each subregion in the resulting ordered sequence. Then, we sample a pivot uniformly at random from $[0, 1)$, and return the first subregion in the ordered sequence for which the cumulative probability is greater than or equal to the pivot. Note that this procedure may fail: This occurs if the pivot is greater than the total cumulative probability.

To sample an argument pair (α_d, α_r) from the subregion, we first sample a point $(\alpha'_d, \alpha'_r) \in \mathbb{Z}^2$ uniformly at random from the subregion. Then, we map (α'_d, α'_r) to the closest admissible argument pair $(\alpha_d, \alpha_r) \in L^\alpha$ by reducing the basis for L^α given in Definition 4.3 and applying Babai's algorithm [1].

To sample an integer pair (j, k) from the distribution, we first sample (α_d, α_r) as described above, and then sample (j, k) uniformly at random from the set of all integer pairs (j, k) yielding (α_d, α_r) using Lemma 4.5. More specifically, we first sample an integer t_r uniformly at random from the set of all admissible values for t_r and then compute (j, k) from (α_d, α_r) and t_r as described in Lemma 4.5.

6 The classical post-processing algorithms

In this section, we describe how d and r are classically recovered from a set $\{(j_1, k_1), \dots, (j_n, k_n)\}$ of pairs produced by performing n independent runs of the quantum algorithm.

6.1 Recovering d from a set of n pairs

To recover d , we exactly follow [7], and use the set of n pairs to form a vector

$$\mathbf{v}_d^k = (\{-2^m k_1\}_{2^{m+\ell}}, \dots, \{-2^m k_n\}_{2^{m+\ell}}, 0) \in \mathbb{Z}^D$$

and a D -dimensional integer lattice L^j with basis matrix

$$\begin{bmatrix} j_1 & j_2 & \cdots & j_n & 1 \\ 2^{m+\ell} & 0 & \cdots & 0 & 0 \\ 0 & 2^{m+\ell} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 2^{m+\ell} & 0 \end{bmatrix}$$

where $D = n + 1$. For some constants $m_1, \dots, m_n \in \mathbb{Z}$, the vector

$$\mathbf{u}_d^j = (\{dj_1\}_{2^{m+\ell}} + m_1 2^{m+\ell}, \dots, \{dj_n\}_{2^{m+\ell}} + m_n 2^{m+\ell}, d) \in L^j$$

is such that the distance

$$\begin{aligned} R_d = |\mathbf{u}_d^j - \mathbf{v}_d^k| &= \sqrt{\sum_{i=1}^n (\{dj_i\}_{2^{m+\ell}} + m_i 2^{m+\ell} - \{-2^m k_i\}_{2^{m+\ell}})^2 + d^2} = \sqrt{\sum_{i=1}^n \underbrace{\{dj_i + 2^m k_i\}_{2^{m+\ell}}^2}_{\alpha_{d,i}^2} + d^2} \\ &= \sqrt{\sum_{i=1}^n \alpha_{d,i}^2 + d^2}. \end{aligned}$$

To recover d , it hence suffices to find $\mathbf{u}_d^j$ by enumerating all vectors in L^j within a D -dimensional hypersphere of radius R_d centered on $\mathbf{v}_d^k$. Its volume is

$$V_D(R_d) = \frac{\pi^{D/2}}{\Gamma(\frac{D}{2} + 1)} R_d^D$$

where Γ is the Gamma function, whilst the fundamental parallelepiped in L^j , that by definition contains a single lattice vector, is of volume $\det L^j = 2^{(m+\ell)n}$.

Heuristically, the hypersphere is hence expected to contain approximately $v_d = V_D(R_d) / \det L^j$ lattice vectors. The exact number depends on the placement of the hypersphere in $\mathbb{Z}^D$, and on the shape of the fundamental parallelepiped in L^j .

6.1.1 Estimating the minimum n required to solve for d

The radius R_d depends on (j_i, k_i) via $\alpha_{d,i}$ for $1 \leq i \leq n$. For fixed n and probability q_d , we exactly follow [7] and estimate the minimum radius $\tilde{R}_d$ such that

$$\Pr \left[R_d := \sqrt{\sum_{i=1}^n \alpha_{d,i}^2 + d^2} \leq \tilde{R}_d \right] \geq q_d \quad (10)$$

by sampling $\alpha_{d,i}$ from the probability distribution. For details on how the estimate is computed, see Section 6.3. It follows from (10) that

$$\Pr \left[v_d := \frac{V_D(R_d)}{\det L^j} \leq \frac{V_D(\tilde{R}_d)}{2^{(m+\ell)n}} \right] \geq q_d. \quad (11)$$

This provides a heuristic bound on the number of lattice vectors v_d that at most have to be enumerated to solve for d , and that holds with probability at least q_d .

6.1.2 Selecting n and solving for d

A simple strategy when solving for d is to select n as described in Section 6.1.1 such that v_d is below a bound equal to the maximum number of vectors that it is computationally feasible to enumerate with probability q_d . This strategy minimizes n at the expense of performing potentially computationally expensive post-processing.

Another strategy is to select n such that $v_d < 2$ with probability q_d , so that there is only one vector in the hypersphere by the heuristic. In theory, this enables us to find $\mathbf{u}_d^j$ with probability q_d by mapping $\mathbf{v}_d^k$ to the closest vector in L^j without enumerating vectors in L^j . In practice, however, the situation is a bit more complicated as $\mathbf{u}_r^j = (\{rj_1\}_{2^{m+\ell}}, \dots, \{rj_n\}_{2^{m+\ell}}, r) \in L^j$ and this vector is short in L^j by construction. To further complicate matters, $\mathbf{u}_r^j/z$ may be in L^j when r is composite, for z some factor of r , see Section 6.2.3. To recover $\mathbf{u}_d^j$, we therefore first map $\mathbf{v}_d^k$ to the closest vector in L^j , and then add or subtract small integer multiples of the shortest vector in the reduced basis for L^j to find $\mathbf{u}_d^j$. This is efficient, except if r has very many small prime factors, and n is close to one, in which case an additional classical post-processing step may be required, see Section 6.2.5.

Note that this complication arises only for general discrete logarithms. It does not arise in [7] when post-processing short discrete logarithms, as the order then does not enter into the equation. Note furthermore that the fact that the order now does play a part may be leveraged in the post-processing, see the next sections.

6.1.3 Selecting n and solving for d by exhausting subsets

The greatest argument $\alpha_{d,i}$ essentially determines the bound on R_d and hence on v_d . A plausible strategy is therefore to make n runs, but to independently post-process all subsets of $n - t$ pairs from the resulting n pairs, for t a constant.

To select n when using this strategy, we specify a bound B on the number of vectors v_d that we accept to enumerate in each lattice of dimension $n - t + 1$, and follow Section 6.1.1 to select the minimum n respecting this bound with probability at least q_d , including only the smallest $n - t$ arguments $\alpha_{d,i}$ when bounding R_d .

With probability q_d , the post-processing then heuristically requires at most B lattice vectors to be enumerated in at most $\binom{n}{t}$ lattices of dimension $n - t + 1$. Note that t must be small as the binomial coefficient grows rapidly in t .

6.1.4 Optimizations when r is known

Note that when r is known, the argument $\alpha_{r,i} = \{rj_i\}_{2^{m+\ell}}$ is known for $1 \leq i \leq n$, and $\alpha_{r,i}$ provides information on $\alpha_{d,i}$ as the arguments are pairwise correlated. When constructing subsets of $n - t$ pairs from the n pairs (j_i, k_i) , the pairs should be included in ascending order sorted by $|\alpha_{r,i}|$. In general, pairs such that $|\alpha_{r,i}|$ exceeds some bound may be rejected as large $|\alpha_{r,i}|$ identify erroneous runs.

6.2 Recovering r from a set of n pairs

To recover r , we instead use that $\mathbf{u}_r^j = (\{rj_1\}_{2^{m+\ell}}, \dots, \{rj_n\}_{2^{m+\ell}}, r) \in L^j$ is a short vector by construction. More specifically, we use that $\mathbf{u}_r^j$ is within a D -dimensional hypersphere in L^j of radius

$$R_r = |\mathbf{u}_r^j| = \sqrt{\sum_{i=1}^n \underbrace{\{rj_i\}_{2^{m+\ell}}^2}_{\alpha_{r,i}^2} + r^2} = \sqrt{\sum_{i=1}^n \alpha_{r,i}^2 + r^2}$$

centered at the origin. In close analogy with [7] and the previous section, we may recover $\mathbf{u}_r^j$ and hence r by enumerating all vectors in this hypersphere. Heuristically, we expect the hypersphere to contain $v_r = V_D(R_r) / \det L^j$ lattice vectors.

This generalization was hinted at in the pre-print of [8]. Furthermore, it is similar to the method employed by Seifert in [24], where he uses what he refers to as simultaneous Diophantine approximation techniques to generalize Shor's [25] continued fractions expansion-based post-processing to higher dimensions. In the case of Shor's original order-finding algorithm, the fact that the problem of finding a continued fraction may be perceived as a lattice problem is observed in [14].

We prefer to describe the post-processing in terms of a shortest vector problem, as this gives us two lattice problems in the same lattice L^j , and as we may re-use the tools previously introduced to estimate the number of runs n required to solve the problem.

6.2.1 Estimating the minimum n required to solve for r

The radius R_r depends on j_i via $\alpha_{r,i}$ for $1 \leq i \leq n$. For fixed n and probability q_r , we proceed in analogy with [7] and estimate the minimum radius $\tilde{R}_r$ such that

$$\Pr \left[R_r := \sqrt{\sum_{i=1}^n \alpha_{r,i}^2 + r^2} \leq \tilde{R}_r \right] \geq q_r \quad (12)$$

by sampling $\alpha_{r,i}$ from the probability distribution. For details on how the estimate is computed, see Section 6.3. It follows from (12) that

$$\Pr \left[v_r := \frac{V_D(R_r)}{\det L^j} \leq \frac{V_D(\tilde{R}_r)}{2^{(m+\ell)n}} \right] \geq q_r. \quad (13)$$

This provides a heuristic bound on the number of lattice vectors v_r that at most have to be enumerated to solve for r , and that holds with probability at least q_r .

6.2.2 Selecting n and solving for r

One strategy when solving for r is to use the heuristic to select n such that v_r is below a bound equal to the maximum number of vectors that it is computationally feasible to enumerate, with probability q_r . This strategy minimizes n at the expense of performing potentially computationally expensive post-processing.

Another strategy is to select n such that $v_r < 2$ with probability q_r , so that there is only one lattice vector in the hypersphere by the heuristic. In theory, this enables us to find $\mathbf{u}_r^j$ with probability q_r by computing the shortest non-zero vector in L^j .

In practice, the heuristic is good when r is prime, as is typically the case when computing discrete logarithms in cryptographic settings. If r is composite, the heuristic is still good, but it may be necessary to perform a small search to find r if r has one or more small prime factors, see Section 6.2.3. If r has many small prime factors, and n is close to one, an additional classical post-processing step may be required to solve efficiently for r , as there may then exist an artificially short non-zero vector in L^j . This additional step is described in Section 6.2.4.

A third strategy is to independently post-process subsets of the pairs output by the quantum computer, in analogy with the procedure described in Section 6.1.3.

6.2.3 Handling composite r

Assume that r is composite. Let $\gcd(\alpha_{r,1}, \dots, \alpha_{r,n}, r) = 2^{k_r} o$ for o odd. Let t be the greatest integer on $[0, k_r]$ for which

$$\alpha_{r,i}/(2^t o) = \{rj_i\}_{2^{m+\ell}}/(2^t o) = \{rj_i/(2^t o)\}_{2^{m+\ell}}$$

for all $i \in [1, n]$. Then $|\mathbf{u}_r^j|/(2^t o) \in L^j$ and $|\mathbf{u}_r^j|/(2^t o) \leq |\mathbf{u}_r^j|$, so $\mathbf{u}_r^j/(2^t o)$ and $r/(2^t o)$ will likely be recovered in the post-processing instead of $\mathbf{u}_r^j$ and r .

For q an odd prime divisor of r , the probability of q also dividing $\alpha_{r,i}$ for all $i \in [1, n]$ is approximately q^{-n} . This implies that r may in general be recovered from $r/(2^t o)$ by exhausting t and o , as the search space is

expected to be small: It is only if r has very many small odd prime divisors, and if n is close to one, that problems may potentially arise. Such problematic cases may be handled efficiently by introducing an additional classical post-processing step, see the next section.

6.2.4 Handling partially smooth r

Let $\mathcal{P}$ be the set of all prime factors $\leq cm$, for $c \geq 1$ some small constant, and let ν_q be the greatest integer such that $q^{\nu_q} < 2^m$. Furthermore, for $r' = r/(2^t o)$ let

$$\tilde{g} = \left[\prod_{q \in \mathcal{P}} q^{\nu_q} \right] g \quad \text{and} \quad \tilde{g}_f = \left[r' \prod_{q \in \mathcal{P} \setminus \{f\}} q^{\nu_q} \right] g \quad \text{for } f \in \mathcal{P},$$

where bracket notation is used to denote generalized exponentiations. Computing $\tilde{g}$ requires at most $2m \cdot \#\mathcal{P} \leq 2cm^2$ group operations¹ to be evaluated classically, for $\#\mathcal{P}$ the cardinality of $\mathcal{P}$. It may hence be done efficiently.

As previously explained, when r is partially very smooth, the classical post-processing algorithm is likely to return $r' = r/(2^t o)$, where o may be large, but where all prime factors of o are small. Assume that all prime factors of $2^t o$ are $\leq cm$. It must then be that $[r'] \tilde{g} \equiv 1$, enabling us to quickly test if r' is on said form. Once $r' = r/(2^t o)$ is found, it is easy to find r : For all $f \in \mathcal{P}$, compute $\tilde{g}_f$ and find the smallest non-negative integer e_f such that $[f^{e_f}] \tilde{g}_f \equiv 1$. Then

$$2^t o = \prod_{f \in \mathcal{P}} f^{e_f}$$

allowing $r = 2^t o \cdot r'$ to be recovered. Computing $\tilde{g}_f$ requires at most $2cm^2$ group operations for each $f \in \mathcal{P}$, for a total of at most $2cm^2 \cdot \#\mathcal{P} \leq 2c^2 m^3$ group operations. The recovery procedure is hence efficient. Note that the procedure may be optimized in various ways. The above description conveys the basic idea.

6.2.5 Computing discrete logarithms when r is partially smooth

If r is partially very smooth, it may be hard to determine d , as there may exist an artificially short vector $|\mathbf{u}_d^j|/2^t o \in L^j$, where o is smooth. Note however that it is still possible to determine $d \bmod r'$, by reducing the last component of the vector $\mathbf{u}_d^j$ sought for in the classical post-processing algorithm modulo $r' = r/2^t o$.

Provided we classically solve the discrete logarithm problem in the residual subgroups of small prime power orders f^{e_f} dividing $2^t o$, which can be done efficiently, the full logarithm d may then be found via the Chinese remainder theorem. This was originally observed by Pohlig and Hellman [20].

6.3 Estimating $\tilde{R}_d$ and $\tilde{R}_r$

To estimate $\tilde{R}_d$ and $\tilde{R}_r$ for m, s and n , known d and r , and a given target success probability q_d or q_r , we exactly follow [7] and sample N sets of n argument pairs $\{(\alpha_{d,1}, \alpha_{r,1}), \dots, (\alpha_{d,n}, \alpha_{r,n})\}$ from the probability distribution. For each set, we compute R_d , sort the resulting list of values in increasing order, and select the value at index $\lfloor (N-1)q_d \rfloor$ to arrive at our estimate for $\tilde{R}_d$. The estimate of $\tilde{R}_r$ is then computed analogously. The constant N controls the accuracy. If N to be sufficiently large in relation to q_d and q_r , and to the variance in the arguments, we expect this approach to yield sufficiently good estimates.

If we fail to sample one or more argument pairs in a set, we closely follow [7] and over-estimate $\tilde{R}_d$ and $\tilde{R}_r$ by letting $R_d = R_r = \infty$ for the set. The entries for the failed sets will then all be sorted to the end of the lists. If the value of $\tilde{R}_d$ or $\tilde{R}_r$ selected from the sorted lists is ∞ , no estimate is produced.

¹ when using the square-and-multiply or double-and-add approach to exponentiation

Let p be the total probability mass covered by the histogram. The probability of all n pairs in a set being in regions covered by the histogram is then p^n . When sampling N sets, the expected number of sets with finite R_d and R_r is Np^n . As Nq_d and Nq_r entries, respectively, in the two lists must be finite for the algorithm to produce an estimate, it follows that it is required that $q_d, q_r > p^n$, with some margin to account for the sampling variance, for estimates to be produced.

7 Estimating the number of runs required

We now have the necessary framework in place to compute concrete estimates for the number of runs n required to attain a given minimum success probability q when recovering both d and r for tradeoff factor s for specific problem instances.

In this section, we describe and exemplify the procedure by computing estimates for a set of concrete problem instances selected in a randomized manner. We furthermore consider how to conjecturally obtain worst case estimates of n .

7.1 Estimating n

To estimate n for a problem instance given by d and r , and for tradeoff factor s , we proceed as follows:

For $n = s + 1, s + 2, \dots$ we first estimate $\tilde{R}_d$ and $\tilde{R}_r$ by sampling $N = 10^6$ sets of n argument pairs (α_d, α_r) , as explained in Section 6.3. We stop and record the smallest n for which the volume quotients $v_d < 2$ and $v_r < 2$ with probability q_d and q_r , respectively, where $q_d, q_r \geq q \geq 99\%$. As the volume quotients each decrease by approximately a constant factor for every increment in n , the minimum n may in practice be found efficiently by interpolation once a few quotients have been computed.

For selected problem instances, we verify the above initial estimate of n by simulating the quantum algorithm and post-processing the simulated output. More specifically, with the initial estimate of n as our starting point, we sample $M = 10^3$ sets of n pairs (j, k) , as explained in Section 5.3, and test whether recovery of both d and r is successful for at least $\lceil Mq \rceil$ sets when executing the post-processing algorithms in Sections 6.1 and 6.2 without enumerating L^j . Depending on the outcome of the test, we either increment or decrement n , and repeat the process, recursively, until the smallest n such that the test passes has been identified. We record this n alongside the initial estimate of n .

In practice, we compute the closest vector in L^j by reducing the lattice basis and applying Babai's [1] nearest plane algorithm. The shortest non-zero vector in L^j is the shortest non-zero vector in the reduced basis. Enumeration is performed using Kannan's [12] original approach, as this is sufficient for our purposes. Note however that there are more efficient approaches in the literature.

To reduce the basis, we closely follow [7] and employ LLL and BKZ [15, 16, 22, 23], as implemented in fpLLL v5.0, with default parameters and a block size of $\min(n + 1, 10)$ for all combinations of m, s and n . We first compute a LLL reduction. If it yields no solution, we proceed to compute a BKZ reduction.

7.2 Selecting m and s

Since the cost of estimating n for a given problem instance is non-negligible, we seek to minimize the number of problem instances considered, whilst selecting problem instances that are representative of those that underpin the currently most widely deployed asymmetric cryptosystems.

To this end, for $m \in \{128, 256, 384, 512, 1024, \dots, 8192\}$, we pick a single combination of d and r using the method described in Section 7.3, and estimate n for a subset of tradeoff factors $s \in \{1, 2, \dots, 8, 10, 20, \dots, 50, 80\}$, such that the total approximation error is negligible.

In terms of group size, the above choices of m capture most currently widely deployed elliptic curve groups, Schnorr groups and safe-prime groups.

7.3 Selecting d and r given m

For each value of m , we select d and r such that $2^{m-1} < d < r < 2^m$ in a randomized manner.

For as long as d and r do not have very special properties, such as being divisible by large powers of two or being otherwise smooth, the exact values of d and r are of no great significance, however: It is the size of d in relation to r , and the size of r in relation to 2^m , that primarily affects the appearance of the distribution and hence the estimates. To avoid having to tabulate d and r for the m we consider, we read d and r from the decimal expansion of Catalan's constant

$$G = \sum_{i=0}^{\infty} \frac{(-1)^i}{(2i+1)^2} = \frac{1}{1^2} - \frac{1}{3^2} + \frac{1}{5^2} - \frac{1}{7^2} + \cdots.$$

Specifically, we let $c_{m,k} = \sum_{j=0}^{m-2} 2^{m-2-j} g_{8191k+j}$ for g_i the i^{th} bit in the decimal expansion of G , and select $r = 2^{m-1} + c_{m,0}$ and $d = 2^{m-1} + (c_{m,1} \bmod c_{m,0})$.

7.4 Experiments and results

The estimates of n in Table 1 were produced by executing the procedure described in the previous sections. As may be seen in the table, n asymptotically tends to $s+1$ as m tends to infinity for fixed s . For fixed m , it holds that $n = s+1$ up to some cutoff point in s .

The estimates are for not enumerating the lattice L^j . By enumerating a bounded number of vectors in the lattice, n may potentially be further reduced. In particular, our experiments show that a single run suffices to solve with probability $q \geq 99\%$ for $s = 1$, provided we accept to enumerate up to $\sim 1.3 \cdot 10^3$ vectors.

Table 1: The estimated number of runs n required to solve for both a general discrete logarithm d and group order r , selected as described in Section 7.3, with $\geq 99\%$ success probability and without enumerating the lattice. For details, see Section 7.4. For A the initial and B the simulated estimate, we print B / A, unless B = A; we then only print A. Dash indicates no estimate. For ϵ the total approximation error, an asterisk indicates that $10^{-4} \leq \epsilon < 10^{-3}$. For all other estimates $\epsilon < 10^{-4}$.

	group and logarithm size m							
	128	256	384	512	1024	2048	4096	8192
tradeoff factor s	1	2	2	2	2	2	2	2
	2	* 3	3	3	3	3	3	3
	3	–	4	4	4	4	4	4
	4	–	* 5	5	5	5	5	5
	5	–	–	6	6	6	6	6
	6	–	–	* 7	7	7	7	7
	7	–	–	–	8	8	8	8
	8	–	–	–	* 10	9	9	9
	10	–	–	–	–	11	11	11
	20	–	–	–	–	–	22	21
	30	–	–	–	–	–	* 35	33 / 32
	40	–	–	–	–	–	–	44
	50	–	–	–	–	–	–	57
	80	–	–	–	–	–	–	54 / 53
								– / 88

As may furthermore be seen in the table, the initial estimates of n are in general verified by the simulations. In general $v_d > v_r$. Hence v_d determines the initial estimate for n . Note however that when the heuristic estimate of v_d is close to two, minor discrepancies between the initial estimates and the simulations may arise.

This phenomenon is discussed in [7]: For large tradeoff factors s in relation to m , increasing or decreasing n typically has a small effect on v_d and v_r . This may lead to slight instabilities in the estimates, as v_d may be

close to two for several values of n . Discrepancies may also arise, especially for large n , if we fail to find the closest and shortest non-zero vectors in L^j , or if sampling fails. Such discrepancies may be amplified by the difference in the sample sizes N and M .

7.5 Generalizing the results to compute worst case estimates

Recall from Section 5.2 that the larger d is permitted to grow in relation to r , and the larger r is permitted to grow in relation to 2^m , the harder it in general becomes to solve for d and r , assuming neither d nor r to be divisible by large powers of two. This may be seen by computing estimates for various combinations of d and r , or simply by examining the appearance of the two-dimensional distribution for various d and r .

Furthermore, recall from Section 5.2 that the marginal distributions along the α_d and α_r axes in the two-dimensional distribution for general discrete logarithms may be seen to correspond to the distributions induced by the quantum algorithms for computing short discrete logarithms with tradeoffs, and orders with tradeoffs, respectively. Both correspondences may be observed numerically by comparing distributions computed for specific problem instances. The correspondence in α_r may furthermore be shown analytically, see Appendix D.

As $v_d > v_r$ in general when d is large in relation to r , and neither d nor r are divisible by large powers of two, we therefore in general expect estimates of n for computing the general discrete logarithm d in a group of m bit order r to agree with estimates of n for computing the short discrete logarithm d , when m is taken as the upper bound on the bit length of d . For d and r selected as in Section 7.3, this correspondence may be observed in practice by comparing the estimates in Table 1 to those in Table B1 in Appendix B.

The above implies that we may conjecturally claim to obtain worst case estimates of n for general discrete logarithms by computing estimates of n for maximal short discrete logarithms $d = 2^m - 1$ when m is taken as an upper bound on the bit length of d . Such estimates are provided in Table B2 in Appendix B.

Note that when comparing estimates of n for general discrete logarithms to those for short discrete logarithms and orders, the comparison must of course be restricted to combinations of m and s such that the total approximation error is negligible. It is reasonable to presume that the correspondence between the distributions would continue to hold even if s was to be permitted to grow a bit past the point where the total approximation error becomes non-negligible for a given m , since the error bound is by no means tight. However, we can not demonstrate the correspondence for such m and s .

8 Order finding with tradeoffs

The algorithm for computing general discrete logarithms in this paper does not require the group order to be known, as neither the quantum algorithm nor the classical post-processing algorithm makes explicit use of the order. If the order of the group is unknown, it may be computed from the same set of pairs (j, k) output by the quantum computer as is used to compute the logarithm.

This implies that the algorithm may be used as an order-finding algorithm. When only the order is of interest, only j need to be computed, as k is not used by the post-processing algorithm that recovers the order. The second stage of the quantum algorithm where k is computed need therefore not be executed when the goal is to perform order finding. If the second stage is removed, the quantum algorithm reduces to the algorithm proposed by Seifert [24]. For $s = 1$ this algorithm in turn reduces to Shor's order-finding algorithm.

This provides a link between our works on computing discrete logarithms, Seifert's work on order finding, and Shor's original work. As for post-processing, Seifert generalizes Shor's continued fractions-based post-processing algorithm to higher dimensions. We instead use lattice-based post-processing.

In Appendix A, we provide a description of Shor's and Seifert's quantum algorithms for order finding, a complete analysis of the probability distributions that they induce, and estimates for the number of runs n required to solve various problem instances for r when using lattice-based post-processing.

9 Summary and conclusion

We generalize and bridge our earlier works on computing short discrete logarithms with tradeoffs, Seifert's work on computing orders with tradeoffs and Shor's groundbreaking works on computing orders and general discrete logarithms. In particular, we enable tradeoffs when computing general discrete logarithms.

Compared to Shor's algorithm for computing general discrete logarithms, this yields a reduction by up to a factor of two in the number of group operations evaluated quantumly in each run, at the expense of having to perform multiple runs. The runs are independent, and may hence be executed in parallel.

Unlike Shor's algorithm, our algorithm does not require the group order to be known. It simultaneously computes both the order and the logarithm. This allows it to outperform Shor's original algorithms with respect to the overall number of group operations that need to be evaluated quantumly in some cases even when not making tradeoffs. One cryptographically relevant example of such a case is the computation of discrete logarithms in Schnorr groups of unknown order.

We analyze the probability distributions induced by our algorithm, and by Shor's and Seifert's order-finding algorithms, describe how all of these algorithms may be simulated when the solution is known, and estimate the number of runs required for a given minimum success probability when making different tradeoffs.

When solving using lattice-based post-processing without enumerating L^j , the number of runs n required for a fixed tradeoff factor s tends to $s + 1$ asymptotically as m tends to infinity. By enumerating, n may be further reduced. Notably, when not making tradeoffs, a single run suffices to solve with at least 99% success probability, provided a small number of lattice vectors are enumerated.

Throughout this work, we have assumed the bit length of r to be known. However, it should be straightforward to generalize the analysis to handle situations where only an upper bound on the bit length of r is known.

Acknowledgement: I am grateful to Johan Håstad for valuable comments and advice, and to Lennart Brynielson and other colleagues for their comments on early versions of this manuscript. Funding and support for this work was provided by the Swedish NCSA that is a part of the Swedish Armed Forces. Computations were performed at PDC at KTH. Access was provided by SNIC.

References

- [1] L. Babai, On Lovász' lattice reduction and the nearest lattice point problem, *Combinatorica* 6 (1986), no. 1, 1–13.
- [2] E. Barker et al., Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography, NIST SP 800-56A (2018), rev. 3.
- [3] W. Diffie and M.E. Hellman, New Directions in Cryptography, *IEEE Transactions on Information Theory* 22 (1976), no. 6, 644–654.
- [4] G. Einarsson, Probability Analysis of a Quantum Computer, *ArXiv quant-ph/0303074* (2003).
- [5] M. Ekerå, Modifying Shor's algorithm to compute short discrete logarithms, *IACR ePrint Archive Report 2016/1128* (2016).
- [6] M. Ekerå, On completely factoring any integer efficiently in a single run of an order finding algorithm, *ArXiv 2007.10044* (2020).
- [7] M. Ekerå, On post-processing in the quantum algorithm for computing short discrete logarithms, *Des. Codes, Cryptogr.* 88 (2020), no. 11, 2313–2335.
- [8] M. Ekerå and J. Håstad, Quantum algorithms for computing short discrete logarithms and factoring RSA integers. In: Lange T., Takagi T. (Eds) *Post-Quantum Cryptography, PQCrypto 2017, LNCS* (2017), no. 10346, 347–363.
- [9] D. Gillmor, Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for Transport Layer Security (TLS), *RFC 7919* (2016).
- [10] R.B. Griffiths and C.-S. Niu, Semiclassical Fourier Transform for Quantum Computation, *Phys. Rev. Lett.* 76 (1996), no. 17, 3228–3231.
- [11] J. Håstad, A. Schiffrin and A. Shamir, The Discrete Logarithm Modulo a Composite Hides $O(n)$ bits, *J. Comput. Syst. Sci.* 47 (1993), no. 3, 376–404.
- [12] R. Kannan, Improved algorithms for integer programming and related lattice problems. In: *Proceedings of the 15th Symposium on the Theory of Computing, STOC '83* (1983), 99–108.

- [13] T. Kivinen and M. Kojo, More Modular Exponentiation (MODP) Diffie-Hellman groups for Internet Key Exchange, RFC 3526 (2003).
- [14] A. Koennecke and P. Wocjan, Recovering the period in Shor's algorithm with Gauss' algorithm for lattice basis reduction, ArXiv 1210.3003 (2013).
- [15] A. Korkine and G. Zolotareff, Sur les formes quadratiques, Math. Ann. 6 (1873), no. 3, 366–389.
- [16] H.W. Lenstra, A.K. Lenstra and L. Lovász, Factoring Polynomials with Rational Coefficients, Math. Ann. 261 (1982), no. 4, 515–534.
- [17] G.L. Miller, Riemann's hypothesis and tests for primality, J. Comput. Syst. Sci. 13 (1976), no. 3, 300–317.
- [18] M. Mosca and A. Ekert: The Hidden Subgroup Problem and Eigenvalue Estimation on a Quantum Computer. In: Proceeding from the First NASA International Conference, QCC '98, LNCS (1999), no. 1509, 174–188.
- [19] S. Parker and M.B. Plenio, Efficient Factorization with a Single Pure Qubit and $\log N$ Mixed Qubits, Phys. Rev. Lett. 85 (2000), no. 14, 3049–3052.
- [20] S.C. Pohlig and M.E. Hellman, An Improved Algorithm for Computing Logarithms over $GF(p)$ and Its Cryptographic Significance, IEEE Trans. Inf. Theory 24 (1978), no. 1, 106–110.
- [21] R.L. Rivest, A. Shamir and L. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Commun. ACM 21 (1978), no. 2, 120–126.
- [22] C.-P. Schnorr, A hierarchy of polynomial time lattice basis reduction algorithms, Theor. Comput. Sci. 53 (1987), no. 2–3, 201–224.
- [23] C.-P. Schnorr and M. Euchner: Lattice basis reduction: Improved practical algorithms and solving subset sum problems, Math. Program. 66 (1994), no. 1–3, 181–199.
- [24] J.-P. Seifert, Using fewer qubits in Shor's factorization algorithm via simultaneous Diophantine approximation. In: Naccache, D. (ed.) CT-RSA 2001, LNCS (2001), no. 2020, 319–327.
- [25] P.W. Shor, Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings of the 35th Annual Symposium on Foundations of Computer Science, SFCS '94 (1994), 124–134.
- [26] P.W. Shor: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Comput. 26 (1997), no. 5, 1484–1509.

A Order finding with tradeoffs

In this appendix, we recall Shor's [25] and Seifert's [24] order-finding algorithms, analyze the probability distributions they induce, show how they may be simulated, and estimate the number of runs n required to solve for r when using lattice-based post-processing.

A.1 The quantum algorithm

Given a generator g of a finite cyclic group of order r of length $\sim m$ bits, Shor's order-finding algorithm [25] outputs an integer j that yields $\sim m$ bits on r . Seifert [24] enabled tradeoffs in Shor's algorithm by modifying it to yield $\sim m/s$ bits on r in each run, for s the tradeoff factor. For $s = 1$ Seifert's algorithm reverts to Shor's algorithm. This allows us to conveniently describe both algorithms below:

1. Let m be the integer such that $2^{m-1} < r < 2^m$, let $\ell = \lceil m/s \rceil$, and let

$$\Psi = \frac{1}{\sqrt{2^{m+\ell}}} \sum_{a=0}^{2^{m+\ell}-1} |a, 0\rangle.$$

2. Compute $[a]g$ and store the result in the second register to obtain

$$\Psi = \frac{1}{\sqrt{2^{m+\ell}}} \sum_{a=0}^{2^{m+\ell}-1} |a, [a]g\rangle.$$

3. Compute a QFT of size $2^{m+\ell}$ of the first register to obtain

$$\Psi = \frac{1}{2^{m+\ell}} \sum_{a=0}^{2^{m+\ell}-1} \sum_{j=0}^{2^{m+\ell}-1} e^{2\pi i aj/2^{m+\ell}} |j, [a]g\rangle.$$

4. Observe the system to obtain j and $y = [e]g$ where $e = a \bmod r$.

Note that Seifert's interpretation of the advantage of his algorithms is that he saves control qubits. This is not the case when recycling control qubits; see the discussion in Section 2 for a more modern interpretation of the advantage.

A.2 The probability of observing j and y

Above, in step 4, the integer j and element $y = [e]g$ are obtained with probability

$$\frac{1}{2^{2(m+\ell)}} \left| \sum_a \exp \left[\frac{2\pi i}{2^{m+\ell}} aj \right] \right|^2 \quad (\text{A1})$$

where the sum is over all a on $0 \leq a < 2^{m+\ell}$ such that $a \equiv e \pmod{r}$.

In this section, we seek to place (A1) on closed form. To this end, we first perform a variable substitution to obtain a contiguous summation interval. As all a that fulfill the condition that $a \equiv e \pmod{r}$ are on the form $a = e + n_r r$ where $0 \leq n_r \leq (2^{m+\ell} - 1 - e)/r$, substituting a for $e + n_r r$ in (A1) and adjusting the phase yields

$$\frac{1}{2^{2(m+\ell)}} \left| \sum_{n_r=0}^{\lfloor (2^{m+\ell}-1-e)/r \rfloor} \exp \left[\frac{2\pi i}{2^{m+\ell}} \alpha_r n_r \right] \right|^2 = \frac{1}{2^{2(m+\ell)}} \left| \sum_{n_r=0}^{\lfloor (2^{m+\ell}-1-e)/r \rfloor} e^{i\theta_r n_r} \right|^2$$

where $\alpha_r = \{rj\}_{2^{m+\ell}}$ and $\theta_r = \theta(\alpha_r) = 2\pi\alpha_r/2^{m+\ell}$. Summing over all e yields

$$\frac{1}{2^{2(m+\ell)}} \sum_{e=0}^{r-1} \left| \sum_{n_r=0}^{\lfloor (2^{m+\ell}-1-e)/r \rfloor} e^{i\theta_r n_r} \right|^2 = \frac{\beta}{2^{2(m+\ell)}} \left| \sum_{n_r=0}^{\lfloor (2^{m+\ell}-1)/r \rfloor} e^{i\theta_r n_r} \right|^2 + \frac{r-\beta}{2^{2(m+\ell)}} \left| \sum_{n_r=0}^{\lfloor (2^{m+\ell}-1)/r \rfloor - 1} e^{i\theta_r n_r} \right|^2 \quad (\text{A2})$$

for β such that $\beta \equiv 2^{m+\ell} \pmod{r}$, as for all e on $0 \leq e < \beta$ it then holds that

$$\left\lfloor (2^{m+\ell} - 1 - e)/r \right\rfloor = \left\lfloor (2^{m+\ell} - 1)/r \right\rfloor$$

whereas for all e on $\beta \leq e < r$, it holds that

$$\left\lfloor (2^{m+\ell} - 1 - e)/r \right\rfloor = \left\lfloor (2^{m+\ell} - 1)/r \right\rfloor - 1.$$

Note that $\beta \not\equiv 0 \pmod{r}$ since we require r to be on $2^{m-1} < r < 2^m$.

A.2.1 Closed-form expressions

Assuming $\theta_r \neq 0$, we may write (A2) on closed form as

$$\frac{\beta}{2^{2(m+\ell)}} \left| \frac{e^{i\theta_r \left(\left\lfloor (2^{m+\ell}-1)/r \right\rfloor + 1 \right)} - 1}{e^{i\theta_r} - 1} \right|^2 + \frac{r-\beta}{2^{2(m+\ell)}} \left| \frac{e^{i\theta_r \left\lfloor (2^{m+\ell}-1)/r \right\rfloor} - 1}{e^{i\theta_r} - 1} \right|^2.$$

Otherwise, if $\theta_r = 0$, we may write (A2) on closed form as

$$\frac{\beta}{2^{2(m+\ell)}} \left(\left\lfloor (2^{m+\ell}-1)/r \right\rfloor + 1 \right)^2 + \frac{r-\beta}{2^{2(m+\ell)}} \left(\left\lfloor (2^{m+\ell}-1)/r \right\rfloor \right)^2.$$

This step of the analysis is similar to a step in the analysis of Einarsson [4].

A.3 Distribution of integers j

In this section we analyze the distribution of integers j that yield α_r .

Definition A.1. Let κ_r denote the greatest integer such that 2^{κ_r} divides r .

Definition A.2. An argument α_r is admissible if there exists an integer j on $0 \leq j < 2^{m+\ell}$ such that $\alpha_r = \{rj\}_{2^{m+\ell}}$.

Claim A.3. All admissible arguments $\alpha_r = \{rj\}_{2^{m+\ell}}$ are multiples of 2^{κ_r} .

Proof. As $2^{\kappa_r} \mid r$ and the modulus is a power of two the claim follows. $\square$

Lemma A.4. The set of integers j on $0 \leq j < 2^{m+\ell}$ that yield the admissible argument α_r is given by

$$j = \left(\frac{\alpha_r}{2^{\kappa_r}} \left(\frac{r}{2^{\kappa_r}} \right)^{-1} + 2^{m+\ell-\kappa_r} t_r \right) \bmod 2^{m+\ell}$$

as t_r runs through all integers on $0 \leq t_r < 2^{\kappa_r}$. Each admissible argument α_r hence occurs with multiplicity 2^{κ_r} .

Proof. As $\alpha_r \equiv rj \pmod{2^{m+\ell}}$, the lemma follows by solving for j . $\square$

A.4 Simulating the quantum algorithm

In this section, we first construct a high-resolution histogram for the probability distribution induced by the quantum algorithm for known r . We then proceed to sample the histogram to simulate the quantum algorithm.

A.4.1 Constructing the histogram

To construct the histogram, we exactly follow [7]: We divide the argument axis into regions and subregions and integrate the closed-form probability expression numerically in each subregion.

First, we subdivide the negative and positive sides of the argument axis into $30 + \mu$ regions where $\mu = \min(\ell - 2, 11)$. Each region thus formed may be uniquely identified by an integer η_r by requiring that for all α_r in the region

$$2^{|\eta_r|} \leq |\alpha_r| \leq 2^{|\eta_r|+1} \quad \text{and} \quad \text{sgn}(\alpha_r) = \text{sgn}(\eta_r)$$

where $m - 30 \leq |\eta_r| < m + \mu - 1$. Then, we subdivide each region into subregions identified by an integer ξ_r by requiring that for all α_r in the subregion

$$2^{|\eta_r|+\xi_r/2^\nu} \leq |\alpha_r| \leq 2^{|\eta_r|+(\xi_r+1)/2^\nu}$$

for ξ_r an integer on $0 \leq \xi_r < 2^\nu$ and ν a resolution parameter.

For each subregion, we compute the approximate probability mass contained within the subregion by applying Simpson's rule, followed by Richardson extrapolation to cancel the linear error term. Simpson's rule is hence applied $2^\nu(1 + 2)$ times in each region. Each application requires the probability to be computed in up to three points (the two endpoints and the midpoint), for which purpose we use the closed-form expression developed in Section A.2.1.

Note that we should furthermore multiply by the multiplicity of arguments 2^{κ_r} , see Lemma A.4 in Section A.3, and divide by 2^{κ_r} to account for the density of distinct pairs in the region. However, these operations cancel. Note also that this method of constructing the histogram assumes κ_r to be small in relation to m .

To obtain a high degree of accuracy in the tail, we fix to $\nu = 11$ for all regions. This enables us use this histogram as a reference when adaptively selecting the resolution for the two-dimensional histogram in Section 5.1, see Lemma D.1.

A.4.2 Understanding the probability distribution

The probability distribution is plotted on the signed logarithmic argument axis in Figure 4 for $m = 2048$ and $s = 30$, and for r selected as explained in Section 7.3. The regions form two contiguous symmetric areas on the argument axis, as is illustrated in Figure A1. As expected, the distribution plotted is virtually identical to the marginal distribution along the vertical α_r axis in Figure 4.

The probability mass is located in the regions where $|\alpha_r| \sim 2^m$, whereas for random outputs the argument would be of size $\sim 2^{m+\ell}$. Hence, a single run of the quantum algorithm yields $\sim \ell \sim m/s$ bits of information on r .

Figure A1: The probability distribution induced by the order-finding algorithm, computed as in Section A.4.1, for $m = 2048$ and $s = 30$, and for r selected as in Section 7.3. To facilitate printing, the resolution has been reduced in this figure.

A.4.3 Sampling the probability distribution

To sample an argument α_r from the distribution, we exactly follow [7]: We first sample a subregion from the histogram and then sample α_r uniformly at random this subregion, with the restriction that 2^{k_r} must divide α_r so that α_r is admissible. To sample a subregion from the histogram, we order all subregions in the histogram by probability, and compute the cumulative probability up to and including each subregion in the resulting ordered sequence, in analogy with Section 5.3.

Then, we sample a pivot uniformly at random from $[0, 1)$, and return the first subregion in the ordered sequence for which the cumulative probability is greater than or equal to the pivot. The sampling operation fails if the pivot is greater than the cumulative probability of the last subregion in the sequence.

To sample an integer j from the distribution, we first sample an argument α_r and then select an integer j yielding α_r uniformly at random from the set of all such integers using Lemma A.4. More specifically, we first sample an integer t_r uniformly at random on the admissible interval for t_r and then compute j from α_r and t_r as described in Lemma A.4.

A.5 Classical post-processing

The probability distribution induced by the quantum algorithm in Section A.1 is virtually identical to the marginal distribution along the α_r axis in Section 5.1. Hence, the classical post-processing algorithm in Section 6.2 may be used to solve sets of n integers j output by the quantum algorithm in Section A.1 for r .

A.6 Estimating the number of runs required

To estimate n for problem instance given by r , we exactly follow [7]:

For $n = s + 1, s + 2, \dots$ we first estimate $\tilde{R}_r$ by sampling $N = 10^6$ sets of n arguments α_r , as explained in Sections A.4.3 and 6.3, and record the smallest n for which the volume quotient $v_r < 2$ with probability $q_r \geq q \geq 99\%$. With this estimate of n as our starting point, we then sample $M = 10^3$ sets of n integers j , as explained in Section A.4.3, and test whether recovery of r is successful for at least $\lceil Mq \rceil$ sets when executing the post-processing algorithm in Section 6.2 without enumerating L^j . Depending on the outcome of the test, we either increment or decrement n , and repeat the process, recursively, until the smallest n such that the test passes has been identified.

Executing this procedure, for m and s selected as described in Section 7.2, both for r selected as explained in Section 7.3, and for maximal $r = 2^m - 1$, produced the estimates in Table A1 and Table A2, respectively. Note that for A the initial and B the simulated estimate, we print B / A , unless $B = A$, we then only print A. Note furthermore that we have excluded $m = 384$ to reduce the table width.

The tabulated estimates are for not enumerating the lattice L^j . By enumerating a bounded number of vectors in the lattice, n may potentially be further reduced. In particular, our experiments show that a single run suffices to solve with probability $q \geq 99\%$ for $s = 1$, provided we accept to enumerate up to $\sim 3.5 \cdot 10^2$ vectors.

A.7 Applications of order finding to integer factoring

Quantum algorithms for order finding may be used to factor integers, as was first proposed by Shor [25] using a reduction due to Miller [17]. To factor a composite integer N , that is odd and not a perfect prime power, Shor originally proposed to proceed as follows: Pick an integer $g \in (1, N)$ and compute $D = \gcd(g, N)$. If $D \neq 1$, then D is a non-trivial factor of N . In practice, small and moderate size factors of N would typically be removed before attempting to factor N via order finding. Hence, it is unlikely that factors would be found in this manner.

Table A1: The estimated number of runs n required to solve for an order r , selected as in Section 7.3, with $\geq 99\%$ success probability, without enumerating the lattice.

	group size m						
	128	256	512	1024	2048	4096	8192
tradeoff factor s	1	2	2	2	2	2	2
	2	3	3	3	3	3	3
	3	4	4	4	4	4	4
	4	6	5	5	5	5	5
	5	7	6	6	6	6	6
	6	9	8	7	7	7	7
	7	12 / 11	9	8	8	8	8
	8	16 / 15	11	10	9	9	9
	10	- / 25	14	12	11	11	11
	20	-	- / 54	28 / 29	24	22	21
	30	-	-	- / 53	39 / 38	34	32
	40	-	-	-	- / 58	48 / 47	44
	50	-	-	-	-	- / 63	56
	80	-	-	-	-	-	- / 95
							- / 87

Table A2: The estimated number of runs n required to solve for a maximal order $r = 2^m - 1$ with $\geq 99\%$ success probability, without enumerating the lattice.

	group size m						
	128	256	512	1024	2048	4096	8192
tradeoff factor s	1	2	2	2	2	2	2
	2	3	3	3	3	3	3
	3	4	4	4	4	4	4
	4	6	5	5	5	5	5
	5	8 / 7	6	6	6	6	6
	6	9	8	7	7	7	7
	7	11 / 12	9	8	8	8	8
	8	17 / 16	11	10	9	9	9
	10	- / 25	14	12	11	11	11
	20	-	- / 55	30 / 29	23 / 24	22	21
	30	-	-	- / 53	37 / 39	34	32
	40	-	-	-	- / 59	48 / 47	44
	50	-	-	-	-	- / 63	57 / 56
	80	-	-	-	-	-	- / 95
							- / 87

If $D = 1$, then g may be perceived as a generator of a cyclic subgroup $\langle g \rangle \subset \mathbb{Z}_N^*$, and its order r computed using a quantum algorithm for order finding.

As $g^r \equiv 1 \pmod{N}$, it must be that $g^r - 1 \equiv 0 \pmod{N}$. If r is even and $g^{r/2} \not\equiv -1 \pmod{N}$, we have that $g^{r/2} \pm 1 \not\equiv 0 \pmod{N}$, whilst

$$g^r - 1 \equiv (g^{r/2} - 1)(g^{r/2} + 1) \equiv 0 \pmod{N},$$

so non-trivial factors of N may be found by computing $\gcd((g^{r/2} \pm 1), N)$. This reduces the integer factoring problem to an order finding problem.

Shor originally proposed to use this reduction, and to simply re-run the whole algorithm if any of the above requirements are not fulfilled, or if the order-finding algorithm fails to yield r . In [25], Shor lower-bounds

the probability of his order-finding algorithm yielding r in a single run, and of non-trivial factors of N being found given r , so as to obtain a lower bound on the overall success probability. A number of improvements have since been proposed, see the introduction to [6] for an overview.

In this appendix, we have shown that the probability of Shor's original order-finding algorithm yielding r in a single run is very close to one. Furthermore, we have estimated the number of runs required to obtain a similarly high success probability when making tradeoffs in Seifert's order-finding algorithm. In [6], it is shown that any integer N may be completely factored classically into all of its constituent prime factors with very high probability after a single call to an order-finding algorithm. Hence, the estimates we provide of the number of runs required for Shor's and Seifert's order-finding algorithms to yield r are also estimates of the number of runs required to completely factor N via these order-finding algorithms.

A.7.1 Factoring RSA integers

Note that if N is an RSA [21] integer, as is typically the case in cryptographic applications, a more efficient approach to factoring N is to use the algorithm of Ekerå and Håstad [8]. This algorithm reduces the RSA integer factoring problem to a short discrete logarithm problem via [11] and solves this problem quantumly.

As is shown in [7], the quantum part of Ekerå-Håstad's algorithm imposes less requirements on the quantum computer than Shor's or Seifert's order-finding algorithms, in each run and overall, both when making and not making tradeoffs. The probability of recovering the logarithm d is very close to one. The two prime factors of N may then be recovered deterministically from d .

B Short discrete logarithms with tradeoffs

The experiments in Appendix A for order finding are analogous with those for short discrete logarithms in [7]. For completeness, and so as to enable comparisons, we have run additional experiments for short discrete logarithms following [7], both for maximal $d = 2^m - 1$, and for d selected as described in Section 7.2. These experiments produced the estimates in Table B1 and Table B2, respectively. Note that for A the initial and B the simulated estimate, we print B / A , unless $B = A$, we then only print A.

Table B1: The estimated number of runs n required to solve for a short logarithm d , selected as in Section 7.3, with $\geq 99\%$ success probability, without enumerating.

	logarithm size m						
	128	256	512	1024	2048	4096	8192
tradeoff factor s							
1	2	2	2	2	2	2	2
2	3	3	3	3	3	3	3
3	4	4	4	4	4	4	4
4	6	5	5	5	5	5	5
5	8	6	6	6	6	6	6
6	10	8	7	7	7	7	7
7	13	10 / 9	8	8	8	8	8
8	18	11	10	9	9	9	9
10	– / 32	15	12	11	11	11	11
20	–	– / 71	32 / 30	24	22	21	21
30	–	–	– / 60	40	35	33 / 32	31
40	–	–	–	– / 62	50 / 48	44	42
50	–	–	–	–	– / 65	57	54 / 53
80	–	–	–	–	–	– / 97	– / 88

Table B2: The estimated number of runs n required to solve for a maximal short logarithm $d = 2^m - 1$ with $\geq 99\%$ success probability, without enumerating.

	logarithm size m						
	128	256	512	1024	2048	4096	8192
tradeoff factor s							
1	2	2	2	2	2	2	2
2	3	3	3	3	3	3	3
3	4	4	4	4	4	4	4
4	6	5	5	5	5	5	5
5	8	6	6	6	6	6	6
6	10	8	7	7	7	7	7
7	13	9	8	8	8	8	8
8	18	11	10	9	9	9	9
10	– / 32	16 / 15	12	11	11	11	11
20	–	– / 71	31	25 / 24	22	21	21
30	–	–	– / 60	40	35	32	32 / 31
40	–	–	–	– / 62	49 / 48	45 / 44	42
50	–	–	–	–	– / 65	57	54 / 53
80	–	–	–	–	–	– / 97	– / 88

C Soundness of the closed-form approximation

In this appendix, we demonstrate the fundamental soundness of the closed-form approximation to $P(\theta_d, \theta_r)$ that we derived in Section 3. This appendix is rather technical and may be considered to constitute supplementary material.

C.1 Introduction and recapitulation

Recall that by Theorem 3.22, the probability $P(\theta_d, \theta_r)$ of the quantum algorithm yielding (j, k) , with associated angle pair (θ_d, θ_r) , summed over all r group elements $y = [e]g \in \mathbb{G}$, may be approximated by

$$\tilde{P}(\theta_d, \theta_r) = \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} f(\theta_r) g(\theta_d, \theta_r)$$

where we have introduced some new notation in the form of the two functions

$$f(\theta_r) = \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right|^2 \quad g(\theta_d, \theta_r) = \left| \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(2^\sigma \theta_d + \lceil -2^\sigma d/r \rceil \theta_r) t} \right|^2$$

that we shall use throughout this section, and that may both be placed on closed form. The error when approximating $P(\theta_d, \theta_r)$ by $\tilde{P}(\theta_d, \theta_r)$ is bounded by

$$\left| \tilde{P}(\theta_d, \theta_r) - P(\theta_d, \theta_r) \right| \leq \tilde{\epsilon}(\theta_d, \theta_r),$$

again by Theorem 3.22, where the function $\tilde{\epsilon}(\theta_d, \theta_r)$ is defined.

C.1.1 Overview of the soundness argument

In what follows, we demonstrate the fundamental soundness of the above closed-form approximation, by summing $\tilde{P}(\theta_d, \theta_r)$ analytically to show that a large fraction of the probability mass is within a specific region of the plane, and by summing $\tilde{\epsilon}(\theta_d, \theta_r)$ analytically to show that the total approximation error in this region is negligible. Asymptotically, in the limit as m tends to infinity for fixed s , the fraction of the probability mass captured tends to one whilst the error tends to zero. This implies that $\tilde{P}(\theta_d, \theta_r)$ asymptotically captures the probability distribution completely and exactly.

C.2 Preliminaries

Before we proceed as outlined above, we first introduce some preliminaries.

Lemma C.1. *Let $\phi \in \mathbb{R}$ and $\theta(u) = 2\pi u/2^\omega$ for $\omega > 0$ an integer. Then*

$$\sum_{u=0}^{2^{\omega+c-\zeta}-1} \left| \sum_{t=0}^{N-1} e^{i(2^\zeta \theta(u) + \phi)t} \right|^2 = 2^{\omega+c-\zeta} N$$

for integers c, ζ and N such that $c \geq 0$, $0 \leq \zeta < \omega$ and $0 < N \leq 2^{\omega-\zeta}$.

Proof. For any $\phi \in \mathbb{R}$ it holds that

$$\left| \sum_{t=0}^{N-1} e^{i(2^\zeta \phi + \phi)t} \right|^2 = \left(\sum_{t=0}^{N-1} e^{i(2^\zeta \phi + \phi)t} \right) \left(\sum_{t=0}^{N-1} e^{-i(2^\zeta \phi + \phi)t} \right)$$

$$\begin{aligned}
&= \sum_{t=-N+1}^{N-1} (N - |t|) e^{i(2^\zeta \phi + \varphi)t} \\
&= N + \sum_{t=1}^{N-1} (N - t) (e^{i(2^\zeta \phi + \varphi)t} + e^{-i(2^\zeta \phi + \varphi)t}).
\end{aligned}$$

Hence

$$\begin{aligned}
\sum_{u=0}^{2^{\omega+c-\zeta}-1} \left| \sum_{t=0}^{N-1} e^{i(2^\zeta \theta(u)+\varphi)t} \right|^2 &= \sum_{u=0}^{2^{\omega+c-\zeta}-1} \left(N + \sum_{t=1}^{N-1} (N - t) (e^{i(2^\zeta \theta(u)+\varphi)t} + e^{-i(2^\zeta \theta(u)+\varphi)t}) \right) \\
&= 2^{\omega+c-\zeta} N + \sum_{t=1}^{N-1} (N - t) \underbrace{\sum_{u=0}^{2^{\omega+c-\zeta}-1} (e^{i(2^\zeta \theta(u)+\varphi)t} + e^{-i(2^\zeta \theta(u)+\varphi)t})}_{=0}
\end{aligned}$$

as for any integer t on $0 < |t| < N \leq 2^{\omega-\zeta}$ and $\zeta < \omega$, the sum

$$\sum_{u=0}^{2^{\omega+c-\zeta}-1} e^{i(2^\zeta \theta(u)+\varphi)t} = e^{i\varphi t} \frac{e^{i2^\zeta(2\pi/2^\omega)2^{\omega+c-\zeta}t} - 1}{e^{i2^\zeta(2\pi/2^\omega)t} - 1} = e^{i\varphi t} \frac{e^{2\pi i 2^c t} - 1}{e^{2\pi i 2^{\zeta-\omega}t} - 1} = 0$$

as the denominator is non-zero, and so the lemma follows. $\square$

C.2.1 Bounding tail regions

Claim C.2. For Δ and N integers such that $1 < \Delta < N$ it holds that

$$\int_{\Delta}^N \frac{du}{u^2} < \sum_{z=\Delta}^{N-1} \frac{1}{z^2} < \int_{\Delta-1}^{N-1} \frac{du}{u^2} < \frac{1}{\Delta-1} \leq \frac{2}{\Delta}.$$

Proof. As

$$\int_z^{z+1} \frac{du}{u^2} = \frac{1}{z+z^2} < \frac{1}{z^2} < \frac{1}{z^2-z} = \int_{z-1}^z \frac{du}{u^2}$$

for z any integer such that $z > 1$, it follows that

$$\int_{\Delta}^N \frac{du}{u^2} = \sum_{z=\Delta}^{N-1} \left(\int_z^{z+1} \frac{du}{u^2} \right) < \sum_{z=\Delta}^{N-1} \frac{1}{z^2} < \sum_{z=\Delta}^{N-1} \left(\int_{z-1}^z \frac{du}{u^2} \right) = \int_{\Delta-1}^{N-1} \frac{du}{u^2}$$

where, for Δ and N integers on $1 < \Delta < N$, it holds that

$$\int_{\Delta-1}^{N-1} \frac{du}{u^2} = \frac{1}{\Delta-1} - \frac{1}{N-1} < \frac{1}{\Delta-1} \leq \frac{2}{\Delta}$$

and so the claim follows. $\square$

Claim C.3. For any $\phi \in \mathbb{R}$ such that $0 < |\phi| \leq \pi$ it holds that

$$\left| \sum_{t=0}^{N-1} e^{i\phi t} \right|^2 \leq \frac{2^4}{\phi^2}.$$

Proof. As $\phi \neq 0$, we have by Claim C.4 below that

$$\left| \sum_{t=0}^{N-1} e^{i\phi} \right|^2 = \left| \frac{e^{iN\phi} - 1}{e^{i\phi} - 1} \right|^2 \leq \frac{2^2}{|e^{i\phi} - 1|^2} \leq \frac{2^4}{\phi^2}$$

and so the claim follows. $\square$

Claim C.4. $|e^{i\phi} - 1| \geq |\phi|/2$ for any $\phi \in \mathbb{R}$ such that $|\phi| \leq \pi$.

Proof. It suffices to show that $|e^{i\phi} - 1|^2 = 2(1 - \cos \phi) \geq \phi^2/4$ from which the claim follows as $\cos \phi \leq 1 - \phi^2/8$ for any $\phi \in \mathbb{R}$ such that $|\phi| \leq \pi$. $\square$

C.2.2 Intervals of admissible arguments and angles

To facilitate the analysis, we need notation to handle intervals of admissible angles:

Definition C.5. Let $\Theta_r(I)$ be the set of distinct admissible θ_r on the interval I .

Definition C.6. For a fixed admissible θ_r , let $\Theta_d(I, \theta_r)$ be the set of distinct admissible θ_d on the interval I .

C.2.3 Parameterizing the admissible arguments and angles

Furthermore, we need a convenient method for parameterizing the distinct admissible argument pairs (α_d, α_r) , or angle pairs (θ_d, θ_r) .

Claim C.7. The admissible arguments α_d and α_r may be parameterized by

$$\alpha_d(u_d, u_r) = (\delta_r u_r \bmod 2^{m-\gamma}) + 2^{m-\gamma} u_d \quad \alpha_r(u_r) = 2^{\kappa_r} u_r$$

and the corresponding admissible angles θ_d and θ_r may be parameterized by

$$\theta_d(u_d, u_r) = \frac{2\pi}{2^{m+\ell}} \alpha_d(u_d, u_r) \quad \theta_r(u_r) = \frac{2\pi}{2^{m+\ell}} \alpha_r(u_r)$$

for integers $u_d \in [-2^{\ell+\gamma-1}, 2^{\ell+\gamma-1})$ and $u_r \in [-2^{m+\ell-\kappa_r-1}, 2^{m+\ell-\kappa_r-1})$ when not accounting for multiplicity.

Proof. By Lemma 4.4, the admissible argument pairs (α_d, α_r) are vectors in L^α in the region of the plane where $\alpha_d, \alpha_r \in [-2^{m+\ell-1}, 2^{m+\ell-1})$. The parameterization takes u_r times the first row and u_d times second row of the basis matrix for L^α . It furthermore uses the second row to reduce the starting point $\delta_r u_r$ modulo $2^{m-\gamma}$. The claim follows from this analysis. $\square$

C.3 Establishing a baseline

We begin by proving that the sum of $\tilde{P}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the region where $\theta_r \in [-\pi, \pi)$ and $\theta_d \in [-\pi/2^\sigma, \pi/2^\sigma)$, tends to one asymptotically in the limit as m tends to infinity for fixed s .

C.3.1 The inner sum over $g(\theta_d, \theta_r)$

Lemma C.8. For $\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma), \theta_r)$, the inner sum

$$\sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma), \theta_r)} g(\theta_d, \theta_r) = 2^{2(\ell-\sigma)+\gamma}.$$

Proof. The function $g(\theta_d, \theta_r)$ is non-negative and periodic in θ_d for fixed θ_r . It cycles exactly 2^σ times on the interval $\theta_d \in [-\pi, \pi)$, as may be seen in Figure C1 where $g(\theta_d, \theta_r)$ is plotted continuously in θ_d for θ_r fixed to zero. Fixing a different value of θ_r shifts the graph cyclically along the θ_d axis.

Figure C1: The function $g(\theta_d, 0)$ plotted continuously in θ_d on the interval $|\theta_d| \leq \pi$ for $\sigma = 3$ and sample parameters selected to make the figure readable.

This implies that we may parameterize θ_d in u_d and u_r using Claim C.7 and sum $\theta_d(u_d, u_r)$ over any consecutive sequence of $2^{\ell+\gamma-\sigma}$ values of u_d for the fixed u_r given by θ_r to sum over all $\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)$.

By using this approach and Lemma C.1 we obtain

$$\begin{aligned} \sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)} g(\theta_d, \theta_r) &= \sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)} \left| \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(2^\sigma \theta_d + \lceil -2^\sigma d/r \rceil \theta_r) t} \right|^2 \\ &= \sum_{u_d = -2^{\ell+\gamma-\sigma-1}}^{2^{\ell+\gamma-\sigma-1}-1} \left| \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(2^\sigma \theta_d(u_d, u_r) + \lceil -2^\sigma d/r \rceil \theta_r(u_r)) t} \right|^2 \\ &= \sum_{u_d = -2^{\ell+\gamma-\sigma-1}}^{2^{\ell+\gamma-\sigma-1}-1} \left| \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(2^\sigma (2\pi 2^{m-\gamma} u_d / 2^{m+\ell}) + \varphi) t} \right|^2 \\ &= \sum_{u_d=0}^{2^{\ell+\gamma-\sigma}-1} \left| \sum_{t=0}^{2^{\ell-\sigma}-1} e^{i(2\pi u_d / 2^{\ell+\gamma-\sigma} + \varphi) t} \right|^2 = 2^{\ell+\gamma-\sigma} \cdot 2^{\ell-\sigma} \end{aligned}$$

where we have used that we may shift the interval in u_d , and introduced the constant phase

$$\varphi = 2^\sigma (2\pi(\delta_r u_r \bmod 2^{m-\gamma}) / 2^{m+\ell}) + \lceil -2^\sigma d/r \rceil \theta_r(u_r),$$

and so the lemma follows. $\square$

C.3.2 The outer sum over $f(\theta_r)$

Lemma C.9. For $\theta_r \in \Theta_r([-\pi, \pi))$, the outer sum

$$\sum_{\theta_r \in \Theta_r([-\pi, \pi))} f(\theta_r) = 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil.$$

Proof. The function $f(\theta_r)$ is non-negative and periodic in θ_r . It cycles exactly once on the interval $\theta_r \in [-\pi, \pi)$. This implies that we may parameterize θ_r in u_r using Claim C.7, and sum over all $2^{m+\ell-\kappa_r}$ values of u_r to sum

over all $\theta_r \in \Theta_r([-\pi, \pi])$. By using this approach and Lemma C.1, we thus obtain

$$\begin{aligned} \sum_{\theta_r \in \Theta_r([-\pi, \pi])} f(\theta_r) &= \sum_{u_r = -2^{m+\ell-\kappa_r-1}}^{2^{m+\ell-\kappa_r-1}-1} \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r(u_r)n_r} \right|^2 \\ &= \sum_{u_r=0}^{2^{m+\ell-\kappa_r}-1} \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i(2\pi u_r/2^{m+\ell-\kappa_r})n_r} \right|^2 = 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \end{aligned}$$

by using that we may shift the interval in u_r , and so the lemma follows. $\square$

C.3.3 Combined result

Lemma C.10. *The combined sum over all distinct admissible angle pairs (θ_d, θ_r) , in the region where $\theta_d \in [-\pi/2^\sigma, \pi/2^\sigma]$ and $\theta_r \in [-\pi, \pi]$, is*

$$\sum_{\substack{\theta_r \in \Theta_r([-\pi, \pi]) \\ \theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)}} \tilde{P}(\theta_d, \theta_r) = 2^{\gamma-\kappa_r} \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil.$$

Proof. By combining Lemmas C.8 and C.9, we obtain

$$\begin{aligned} \sum_{\substack{\theta_r \in \Theta_r([-\pi, \pi]) \\ \theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)}} \tilde{P}(\theta_d, \theta_r) &= \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} \sum_{\theta_r \in \Theta_r([-\pi, \pi])} f(\theta_r) \sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)} g(\theta_d, \theta_r) \\ &= \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} \cdot 2^{2(\ell-\sigma)+\gamma} \cdot 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil = 2^{\gamma-\kappa_r} \cdot \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \end{aligned}$$

as the inner sum reduces to a constant, and so the lemma follows. $\square$

It follows from Lemma C.10 above that the sum of $\tilde{P}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) in the region where $\theta_d \in [-\pi/2^\sigma, \pi/2^\sigma]$ and $\theta_r \in [-\pi, \pi]$ tends to one as m tends to infinity for fixed s , when accounting for the fact that each distinct admissible angle pair (θ_d, θ_r) occurs with multiplicity $2^{\kappa_r-\gamma}$ by Lemma 4.4.

The total approximation error, as upper-bounded by summing $\tilde{e}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the region, is non-negligible however. In the next section we address this problem by reducing the size of the region.

C.4 Adapting the region to reduce the error

In this section, we show that the sum of $\tilde{P}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region as defined below, captures a fraction of the probability mass in τ .

Definition C.11. The central region is the part of the plane where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, for $B_d = 2^{\tau-\ell+1}\pi$ and $B_r = B_d/2$, and for τ an integer constant such that $1 < \tau < \ell - \sigma - 1$.

In the next section, we describe how the approximation error, as upper-bounded by summing $\tilde{e}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region, depends on τ . For appropriate σ and τ , a large fraction of the probability mass is in the central region, whilst the total approximation error is negligible in the region.

Note that by the above definition of B_d and B_r , all argument pairs (α_d, α_r) such that $|\alpha_d| \leq 2^{m+\tau}$ and $|\alpha_r| \leq 2^{m+\tau-1}$ are in the central region. Note furthermore that $B_r < B_d = 2^{\tau-\ell+1}\pi \leq 2^{-\sigma-1}\pi$, so the central region is a subregion of the region we considered in the previous section.

C.4.1 The inner sum over $g(\theta_d, \theta_r)$

Lemma C.12. For $\theta_r \in \Theta_r([-B_r, B_r])$, the inner sum

$$\sum_{\theta_d \in \Theta_d([-B_d, B_d], \theta_r)} g(\theta_d, \theta_r) \geq 2^{2(\ell-\sigma)+\gamma} \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right).$$

Proof. First observe that for $I_d = [-\pi/2^\sigma, -B_d] \cup [B_d, \pi/2^\sigma]$ we have

$$\sum_{\theta_d \in \Theta_d([-B_d, B_d], \theta_r)} g(\theta_d, \theta_r) \geq 2^{2(\ell-\sigma)+\gamma} - \sum_{\theta_d \in \Theta_d(I_d, \theta_r)} g(\theta_d, \theta_r)$$

as $g(\theta_d, \theta_r)$ is non-negative, and as by dividing the interval

$$\begin{aligned} \sum_{\theta_d \in \Theta_d([- \pi/2^\sigma, \pi/2^\sigma], \theta_r)} g(\theta_d, \theta_r) &= \sum_{\theta_d \in \Theta_d([- \pi/2^\sigma, -B_d], \theta_r)} g(\theta_d, \theta_r) + \\ &\quad \sum_{\theta_d \in \Theta_d([-B_d, B_d], \theta_r)} g(\theta_d, \theta_r) + \\ &\quad \sum_{\theta_d \in \Theta_d([B_d, \pi/2^\sigma], \theta_r)} g(\theta_d, \theta_r) = 2^{2(\ell-\sigma)+\gamma} \end{aligned}$$

where we also used Lemma C.8. We hence seek an upper bound to

$$\sum_{\theta_d \in \Theta_d(I_d, \theta_r)} g(\theta_d, \theta_r) = \sum_{\theta_d \in \Theta_d(I_d, \theta_r)} g(\theta_d + \lceil -2^\sigma d/r \rceil \theta_r/2^\sigma, 0) \leq \sum_{\theta_d \in \Theta_d(I_d, \theta_r)} h(\theta_d + \lceil -2^\sigma d/r \rceil \theta_r/2^\sigma) \quad (C1)$$

that is independent of θ_r , where we have used Claim C.3 to obtain (C1), by introducing the function $h(\theta_d) = 2^4/(2^\sigma \theta_d)^2$ that is strictly decreasing in $|\theta_d|$.

The situation is depicted in Figure C2, where $g(\theta_d, \theta_r)$ for $\theta_r = 0$ is plotted continuously in θ_d , for $|\theta_d| \leq \pi$ in the top graph, and $|\theta_d| \leq \pi/2^\sigma$ in the middle graph.

Fixing a non-zero $\theta_r \in \Theta_r([-B_r, B_r])$ shifts the top and middle graphs in Figure C2 cyclically by $\lceil -2^\sigma d/r \rceil \theta_r/2^\sigma$. As $|\lceil -2^\sigma d/r \rceil \theta_r/2^\sigma| \leq |\theta_r| \leq B_r$, the maximum cyclic shift in θ_d is upper-bounded by B_r , see the bottom graph in Figure C2 where $g(\theta_d + B_r, 0)$ is plotted in yellow and $g(\theta_d - B_r, 0)$ in green.

To upper-bound (C1) it therefore suffices to sum over all distinct admissible θ_d on $I_r = [-\pi/2^\sigma, -B_r] \cup [B_r, \pi/2^\sigma]$, as this captures all distinct admissible θ_d in the left and right tail regions under any cyclic shift.

We have that

$$\begin{aligned} (C1) &= \sum_{\theta_d \in \Theta_d(I_d, \theta_r)} h(\theta_d + \lceil -2^\sigma d/r \rceil \theta_r/2^\sigma) \\ &\leq \max_{\theta_r \in \Theta_r([-B_r, B_r])} \sum_{\theta_d \in \Theta_d(I_r, \theta_r)} h(\theta_d) \end{aligned} \quad (C2)$$

$$= \sum_{\theta_d \in \Theta_d(I_r, 0)} h(\theta_d) = \sum_{\theta_d \in \Theta_d([B_r, \pi/2^\sigma], 0)} 2h(\theta_d) \quad (C3)$$

due to symmetry, where we have maximized the set of admissible θ_d over θ_r .

Recall that by Lemma 4.4 there is one distinct admissible argument α_d on the interval $[0, 2^{m-\gamma})$ for a given fixed α_r . Hence there is one distinct admissible θ_d on the interval $[0, 2^{-\ell-\gamma+1}\pi)$ for a given fixed θ_r . All other distinct admissible θ_d spread out from the starting point, equidistantly separated by a distance of $2^{-\ell-\gamma+1}\pi$. The distinct admissible θ_d may occur with multiplicity; however all distinct admissible θ_d occur with the same multiplicity, again see Lemma 4.4.

This implies that the sum in (C2) is maximized for θ_r equal to zero, as both endpoints of the interval $B_r \leq |\theta_d| \leq \pi/2^\sigma$ are then admissible, maximizing both the number of distinct admissible θ_d on the interval, and the contribution from each distinct admissible θ_d as $h(\theta_d)$ is strictly decreasing in $|\theta_d|$.

By Claim C.7, the distinct admissible θ_d may be parameterized in u_d and u_r where

$$\theta_d(u_d, u_r) = 2\pi((\delta_r u_r \bmod 2^{m-\gamma}) + 2^{m-\gamma} u_d)/2^{m+\ell}.$$

Figure C2: The functions $g(\theta_d, 0)$ and $h(\theta_d) = 2^4 / (2^\sigma \theta_d)^2$ plotted for $\sigma = 3$, $\ell = 9$ and $\tau = 3$. The maximum cyclic shift is bounded by $B_r = B_d/2$.

Now $\theta_r = 0$ implies $u_r = 0$, which in turn implies $2^{\tau-\ell}\pi = B_d/2 = B_r \leq 2\pi u_d/2^{\ell+\gamma} \leq \pi/2^\sigma$, or more succinctly $2^{\tau+\gamma-1} \leq u_d \leq 2^{\ell+\gamma-\sigma-1}$, which yields

$$\begin{aligned} (C3) &= \sum_{u_d=2^{\tau+\gamma-1}}^{2^{\ell+\gamma-\sigma-1}} 2h(\theta_d(u_d, u_r)) = \sum_{u_d=2^{\tau+\gamma-1}}^{2^{\ell+\gamma-\sigma-1}} \frac{2^5}{(2^\sigma \cdot 2\pi u_d/2^{\ell+\gamma})^2} \\ &= 2^{2(\ell-\sigma+\gamma)} \frac{2^3}{\pi^2} \sum_{u_d=2^{\tau+\gamma-1}}^{2^{\ell+\gamma-\sigma-1}} \frac{1}{u_d^2} \leq 2^{2(\ell-\sigma+\gamma)} \frac{2^3}{\pi^2} \frac{2}{2^{\tau+\gamma-1}} = 2^{2(\ell-\sigma)+\gamma} \frac{2^5}{\pi^2} \frac{1}{2^\tau} \end{aligned}$$

where we have used Claim C.2 and that $\gamma \geq 0$ and $\tau > 1$. This implies

$$\sum_{\theta_d \in \Theta_d([-B_d, B_d], \theta_r)} g(\theta_d, \theta_r) \geq 2^{2(\ell-\sigma)+\gamma} - 2^{2(\ell-\sigma)+\gamma} \frac{2^5}{\pi^2} \frac{1}{2^\tau} = 2^{2(\ell-\sigma)+\gamma} \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right)$$

and so the lemma follows. $\square$

C.4.2 The outer sum over $f(\theta_r)$

Lemma C.13. For $\theta_r \in \Theta_r([-B_r, B_r])$, the outer sum

$$\sum_{\theta_r \in \Theta_r([-B_r, B_r])} f(\theta_r) \geq 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right).$$

Proof. First observe that for $I_r = [-\pi, -B_r] \cup [B_r, \pi]$ it holds that

$$\sum_{\theta_r \in \Theta_r([-B_r, B_r])} f(\theta_r) \geq 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil - \sum_{\theta_r \in \Theta_r(I_r)} f(\theta_r)$$

as $f(\theta_r)$ is non-negative and

$$\sum_{\theta_r \in \Theta_r([- \pi, \pi])} f(\theta_r) = \sum_{\theta_r \in \Theta_r([- \pi, -B_r])} f(\theta_r) + \sum_{\theta_r \in \Theta_r([-B_r, B_r])} f(\theta_r) + \sum_{\theta_r \in \Theta_r([B_r, \pi])} f(\theta_r)$$

where, by Lemma C.9, the left hand sum

$$\sum_{\theta_r \in \Theta_r([- \pi, \pi])} f(\theta_r) = 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil.$$

To prove the lemma, we seek an upper bound to

$$\sum_{\theta_r \in \Theta_r(I_r)} f(\theta_r) \leq \sum_{\theta_r \in \Theta_r(I_r)} \frac{2^4}{\theta_r^2} \leq \sum_{\theta_r \in \Theta_r([B_r, \pi])} \frac{2^5}{\theta_r^2} \quad (C4)$$

where we have used Claim C.3, that $f(\theta_r)$ is symmetric around the origin, and that the distinct admissible θ_r are equidistantly separated by a distance of 2^{κ_r} around the origin by Lemma 4.4. The distinct admissible θ_r may occur with multiplicity; however all distinct admissible θ_r occur with the same multiplicity.

By Claim C.7, the distinct admissible θ_r may be parameterized in u_r where $\theta_r(u_r) = 2\pi(2^{\kappa_r}u_r)/2^{m+\ell}$, which implies $2^{\tau-\ell}\pi = B_r \leq 2\pi(2^{\kappa_r}u_r)/2^{m+\ell} \leq \pi$, or more succinctly $2^{m+\tau-\kappa_r-1} \leq u_r \leq 2^{m+\ell-\kappa_r-1}$, which yields

$$\begin{aligned} (C4) &= \sum_{u_r=2^{m+\tau-\kappa_r-1}}^{2^{m+\ell-\kappa_r-1}} \frac{2^5}{(2\pi 2^{\kappa_r} u_r / 2^{m+\ell})^2} = 2^{2(m+\ell-\kappa_r)} \frac{2^3}{\pi^2} \sum_{u_r=2^{m+\tau-\kappa_r-1}}^{2^{m+\ell-\kappa_r-1}} \frac{1}{u_r^2} \\ &\leq 2^{2(m+\ell-\kappa_r)} \frac{2^3}{\pi^2} \frac{2}{2^{m+\tau-\kappa_r-1}} = 2^{m+2\ell-\kappa_r} \frac{2^5}{\pi^2} \frac{1}{2^\tau} \leq 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \frac{2^5}{\pi^2} \frac{1}{2^\tau} \end{aligned}$$

where we have used Claim C.2 and that $\gamma \geq 0$ and $\tau > 1$. This implies

$$\begin{aligned} \sum_{\theta_r \in \Theta_r([-B_r, B_r])} f(\theta_r) &\geq 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil - 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \frac{2^5}{\pi^2} \frac{1}{2^\tau} \\ &= 2^{m+\ell-\kappa_r} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau} \right) \end{aligned}$$

and so the lemma follows. $\square$

C.4.3 Combined result

Lemma C.14. *The combined sum over all distinct admissible angle pairs (θ_d, θ_r) , in the central region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, is*

$$\sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} \tilde{P}(\theta_d, \theta_r) \geq 2^{\gamma-\kappa_r} \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau} \right)^2.$$

Proof. From Lemmas C.12 and C.13 it follows that

$$\sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} \tilde{P}(\theta_d, \theta_r) = \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} \sum_{\theta_r \in \Theta_r([-B_r, B_r])} f(\theta_r) \sum_{\theta_d \in \Theta_d([-B_d, B_d], \theta_r)} g(\theta_d, \theta_r)$$

$$\begin{aligned}
&\geq \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} 2^{m+\ell-\kappa_r} \cdot 2^{2(\ell-\sigma)+\gamma} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right)^2 \\
&= 2^{\gamma-\kappa_r} \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right)^2
\end{aligned}$$

and so the lemma follows. $\square$

C.5 Main soundness result

In this section, we combine results from the previous sections into our main soundness result.

C.5.1 Bounding the probability mass in the central region

Theorem C.15. *The sum of $\tilde{P}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, is bounded by*

$$\frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right)^2 \leq \sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{\kappa_r - \gamma} \tilde{P}(\theta_d, \theta_r) \leq \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil.$$

Proof. The theorem follows by combining Lemmas C.10 and C.14. $\square$

Theorem C.15 above lower-bounds the fraction of the probability mass that is located within the central region as a function of τ . The fraction of the probability mass that falls outside the central region decreases exponentially in τ .

C.5.2 Bounding the total error in the central region

Theorem C.16. *The total error when approximating $P(\theta_d, \theta_r)$ by $\tilde{P}(\theta_d, \theta_r)$, as upper-bounded by summing $\tilde{e}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, is bounded by*

$$\sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{\kappa_r - \gamma} \tilde{e}(\theta_d, \theta_r) \leq 2^{m+2\tau} D \left(\frac{2^6}{2^\sigma} + \frac{2^5}{2^\ell} \right) + \frac{2^{\tau+\sigma+2}}{2^\ell} \pi \left(1 + \frac{2^{\tau+\sigma}}{2^\ell} \pi \right) \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil$$

where D is the density of admissible angle pairs (θ_d, θ_r) in the region.

Proof. The error when approximating $P(\theta_d, \theta_r)$ by $\tilde{P}(\theta_d, \theta_r)$ is bounded by

$$\tilde{e}(\theta_d, \theta_r) = \frac{2^4}{2^{m+\sigma}} + \frac{2^3}{2^{m+\ell}} + \frac{2^\sigma}{2} (|\theta_d| + |\theta_r|) \left(2 + \frac{2^\sigma}{2} (|\theta_d| + |\theta_r|) \right) \tilde{P}(\theta_d, \theta_r)$$

by Theorem 3.22. We sum $\tilde{e}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, where $B_d = 2^{\tau-\ell+1} \pi$ and $B_r = B_d/2$ by Definition C.11. This is equivalent to summing over all admissible argument pairs (α_d, α_r) , with multiplicity, in the region where $|\alpha_d| \leq 2^{m+\tau}$ and $|\alpha_r| \leq 2^{m+\tau-1}$.

As $m > 0$ and $\tau > 1$ by Definition C.11, the area of this region is

$$\begin{aligned}
(2 \cdot 2^{m+\tau} + 1)(2 \cdot 2^{m+\tau-1} + 1) &= 2^{2(m+\tau)+1} + 2^{m+\tau+1} + 2^{m+\tau} + 1 \\
&\leq 2^{2(m+\tau+1)}
\end{aligned}$$

from which it follows that the region contains at most $2^{2(m+\tau+1)}D$ admissible pairs (θ_d, θ_r) , where D is the density of admissible pairs with multiplicity.

If we furthermore use that $|\theta_d| + |\theta_r| \leq 2^{\tau-\ell+2}\pi$, this implies that

$$\begin{aligned} \sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{\kappa_r - \gamma} \tilde{e}(\theta_d, \theta_r) &\leq 2^{2(m+\tau+1)}D \left(\frac{2^4}{2^{m+\sigma}} + \frac{2^3}{2^{m+\ell}} \right) + 2^{\tau-\ell+\sigma+2}\pi(1 + 2^{\tau-\ell+\sigma}\pi) \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \\ &\leq 2^{m+2\tau}D \left(\frac{2^6}{2^\sigma} + \frac{2^5}{2^\ell} \right) + \frac{2^{\tau+\sigma+2}}{2^\ell}\pi \left(1 + \frac{2^{\tau+\sigma}}{2^\ell}\pi \right) \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil \end{aligned}$$

where we have used that

$$\sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{\kappa_r - \gamma} \tilde{P}(\theta_d, \theta_r) \leq \frac{r}{2^{m+\ell}} \left\lceil \frac{2^{m+\ell}}{r} \right\rceil$$

by Theorem C.15, and so the theorem follows. $\square$

By Lemmas 4.7 and 4.8, the density D of admissible argument pairs (α_d, α_r) , or equivalently angle pairs (θ_d, θ_r) , when accounting for multiplicity, in the region is approximately 2^{-m} for random problem instances. Asymptotically, the density tends to 2^{-m} as m tends to infinity for fixed s .

Furthermore, the density is exactly 2^{-m} in rectangular regions of the plane of side length multiples of $2^{m-\gamma}$ and $2^{m-\gamma+\kappa_r}$ by Lemma 4.9. The region in Theorem C.16 above may be adapted to meet these requirements.

To understand the implications of the above theorem for the upper bound on the total approximation error in the central region, it remains to select σ to minimize the bound.

C.5.3 Selecting σ to minimize the bound on the total error in the central region

To select the integer parameter σ on $0 < \sigma < \ell$ so as to minimize the bound on the total error given in Theorem C.16, we first approximate the error bound by

$$\underbrace{\frac{2^{2\tau+6}}{2^\sigma}}_{\epsilon_1} + \underbrace{\frac{2^{2\tau+5}}{2^\ell}}_{\epsilon_2} + \underbrace{\frac{2^{\tau+\sigma+2}}{2^\ell}\pi}_{\epsilon_3} + \underbrace{\left(\frac{1}{2} \frac{2^{\tau+\sigma+2}}{2^\ell}\pi \right)^2}_{\epsilon_4}$$

where we have used that $D \approx 2^{-m}$ and $(r/2^{m+\ell}) \left\lceil 2^{m+\ell}/r \right\rceil \approx 1$, with equality in the limit as m tends to infinity for fixed s . The approximation is only good when all error terms are less than one, so the term ϵ_3 is greater than $\epsilon_4 = (\epsilon_3/2)^2$. As ϵ_2 does not depend on σ , we hence seek to select σ to equate ϵ_1 and ϵ_3 . This yields

$$\frac{2^{2\tau+6}}{2^\sigma} = \frac{2^{\tau+\sigma+2}}{2^\ell}\pi \quad \Rightarrow \quad \sigma = \left\lfloor \frac{1}{2}(\ell + \tau + 4 - \log_2 \pi) \right\rfloor.$$

If σ is fixed accordingly, the error bound obtained by summing $\tilde{e}(\theta_d, \theta_r)$ analytically over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, is heuristically minimized. For this σ , the two main error terms

$$\epsilon_1 \approx \epsilon_3 \approx \frac{2^{3\tau/2+4}}{2^{\ell/2}}\sqrt{\pi}. \quad (\text{C5})$$

For as long as $2^{3\tau/2+4}\sqrt{\pi}$ is much smaller than $2^{\ell/2}$, we heuristically expect the upper bound on the total error given in Theorem C.16 to be negligible.

C.5.4 Asymptotic soundness results

Theorem C.17. For fixed s and τ , and $\sigma = \lfloor (\ell + \tau + 4 - \log_2 \pi)/2 \rfloor$, the sum of $\tilde{P}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, is bounded by

$$\left(1 - \frac{2^5}{\pi^2} \frac{1}{2^\tau}\right)^2 \leq \lim_{m \rightarrow \infty} \sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{K_r - \gamma} \tilde{P}(\theta_d, \theta_r) \leq 1 \quad (\text{C6})$$

in the limit as m tends to infinity. The error $|P(\theta_d, \theta_r) - \tilde{P}(\theta_d, \theta_r)| \leq \tilde{e}(\theta_d, \theta_r)$ and the sum of $\tilde{e}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region tends to

$$\lim_{m \rightarrow \infty} \sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{K_r - \gamma} \tilde{e}(\theta_d, \theta_r) = 0 \quad (\text{C7})$$

in the same limit.

Proof. The bound in (C6) follows immediately by taking the limit, as m tends to infinity for fixed s and τ , of the bound given in Theorem C.15. Analogously (C7) follows by taking the limit, as m tends to infinity for fixed s and τ , and for σ as in the formulation of this theorem, of Theorem C.16, where D tends to 2^{-m} in the limit by Lemma 4.8, and so the theorem follows. $\square$

The above theorem states that an arbitrarily large constant fraction of the probability mass may be captured asymptotically by expanding the region in τ .

As the bound on the error when approximating $P(\theta_d, \theta_r)$ by $\tilde{P}(\theta_d, \theta_r)$ in the region tends to zero asymptotically, $\tilde{P}(\theta_d, \theta_r)$ equals $P(\theta_d, \theta_r)$ asymptotically in the region. Furthermore, all probability mass is in the region asymptotically when τ tends to infinity with m at a moderated rate. This implies that $\tilde{P}(\theta_d, \theta_r)$ asymptotically captures the probability distribution completely and exactly. Corollary C.18 below formalizes these observations:

Corollary C.18. For fixed s , for $\tau = \lfloor \ell/6 \rfloor$ and $\sigma = \lfloor (\ell + \tau + 4 - \log_2 \pi)/2 \rfloor$, the sum of $\tilde{P}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region where $|\theta_d| \leq B_d$ and $|\theta_r| \leq B_r$, tends to

$$\lim_{m \rightarrow \infty} \sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{K_r - \gamma} \tilde{P}(\theta_d, \theta_r) = 1 \quad (\text{C8})$$

in the limit as m tends to infinity. The error $|P(\theta_d, \theta_r) - \tilde{P}(\theta_d, \theta_r)| \leq \tilde{e}(\theta_d, \theta_r)$ and the sum of $\tilde{e}(\theta_d, \theta_r)$ over all admissible angle pairs (θ_d, θ_r) , with multiplicity, in the central region tends to

$$\lim_{m \rightarrow \infty} \sum_{\substack{\theta_r \in \Theta_r([-B_r, B_r]) \\ \theta_d \in \Theta_d([-B_d, B_d], \theta_r)}} 2^{K_r - \gamma} \tilde{e}(\theta_d, \theta_r) = 0 \quad (\text{C9})$$

in the same limit.

Proof. The bound in (C8) follows immediately by taking the limit as m tends to infinity for fixed s , and for τ as in the formulation of this corollary, of the bound given in Theorem C.15. Analogously (C9) follows by taking the limit, as m tends to infinity for fixed s , and for σ and τ as in the formulation of this corollary, of Theorem C.16, where D tends to 2^{-m} in the limit by Lemma 4.8. This is easy to see, as the two main error terms ϵ_1 and ϵ_3 in (C5) tend to

$$\lim_{m \rightarrow \infty} \frac{2^{3\lfloor \ell/6 \rfloor/2+4}}{2^{\ell/2}} \sqrt{\pi} = \lim_{m \rightarrow \infty} \frac{2^{\ell/4+4}}{2^{\ell/2}} \sqrt{\pi} = \lim_{m \rightarrow \infty} \frac{2^4}{2^{\ell/4}} \sqrt{\pi} = 0,$$

where we may remove the rounding operation in the limit, and as the requirement that $1 < \tau < \ell - \sigma - 1$ in Definition C.11 is respected in the limit. Furthermore $\epsilon_4 < \epsilon_3$ in the limit, and it is easy to see that ϵ_2 tends to zero in the limit. The corollary follows from this analysis. $\square$

D Marginal distributions

By using results and notation from the soundness analysis in Appendix C, we may immediately derive a closed-form expression for the marginal distribution that arises when summing $\tilde{P}(\theta_d, \theta_r)$ over all admissible angles θ_d with multiplicity.

Lemma D.1. *For $\theta_r \in \Theta_r([-\pi, \pi])$, the marginal probability distribution that arises when summing $\tilde{P}(\theta_d, \theta_r)$ over all $\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)$ is*

$$\sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)} \frac{2^{K_r - \gamma}}{2^{K_r}} \tilde{P}(\theta_d, \theta_r) = \frac{r}{2^{2(m+\ell)}} \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right|^2$$

when accounting for multiplicity.

Proof. By Lemma C.8 we have that

$$\begin{aligned} \sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)} \tilde{P}(\theta_d, \theta_r) &= \frac{2^{2\sigma} r}{2^{2(m+2\ell)}} f(\theta_r) \sum_{\theta_d \in \Theta_d([-\pi/2^\sigma, \pi/2^\sigma], \theta_r)} g(\theta_d, \theta_r) \\ &= \frac{2^\gamma r}{2^{2(m+\ell)}} f(\theta_r) = \frac{2^\gamma r}{2^{2(m+\ell)}} \left| \sum_{n_r=0}^{\lceil 2^{m+\ell}/r \rceil - 1} e^{i\theta_r n_r} \right|^2 \end{aligned}$$

from which the lemma follows, as the pairs (θ_d, θ_r) occur with multiplicity $2^{K_r - \gamma}$ by Lemma 4.4, and the angles θ_r with multiplicity 2^{K_r} by Lemma A.4. $\square$

The above expression for the marginal probability distribution is derived from the approximation $\tilde{P}(\theta_d, \theta_r)$. It corresponds to the exact expression derived in Appendix A for the order-finding algorithm with tradeoffs. Note that there are minor differences between the two expressions. These are explained by $\tilde{P}(\theta_d, \theta_r)$ being an approximation to $P(\theta_d, \theta_r)$, whilst the expression in Appendix A is exact.

Another way to understand why this correspondence arises is to observe that when using qubit recycling, j may first be computed, after which k may be computed. At the point in time when j has been computed and read out, but the computation of k has not yet begun, we will have executed Shor's or Seifert's order-finding algorithms. Hence j , α_r and θ_r are distributed as in these algorithms.

A closed-form analytical expression for the marginal distribution that arises when summing over all admissible θ_r is seemingly less straightforward to derive. Numerically, the marginal distribution may however be seen to correspond to that for short logarithms as stated in Section 5.2.