Faster classical boson sampling

Peter Clifford and Raphaël Clifford

1 Department of Statistics University of Oxford United Kingdom
2 Department of Computer Science University of Bristol United Kingdom

E-mail: raphael.clifford@bristol.ac.uk

Keywords: boson sampling, quantum advantage, classical algorithms

Abstract
Since its introduction boson sampling has been the subject of intense study in the world of quantum computing. In the context of Fock-state boson sampling, the task is to sample independently from the set of all \( n \times n \) submatrices built from possibly repeated rows of a larger \( m \times n \) complex matrix according to a probability distribution related to the permanents of the submatrices. Experimental systems exploiting quantum photonic effects can in principle perform the task at great speed. For classical computing, Aaronson and Arkhipov (2011) showed that exact boson sampling problem cannot be solved in polynomial time unless the polynomial hierarchy collapses to the third level. Indeed for a number of years the fastest known exact classical algorithm ran in \( O((m+n-1)n^2^n) \) time per sample, emphasising the potential speed advantage of quantum computation. The advantage was reduced by Clifford and Clifford (2018), who gave a significantly faster classical solution taking \( O(n^{2^n} + \text{poly}(m,n)) \) time and linear space, matching the complexity of computing the permanent of a single matrix when \( m \) is polynomial in \( n \). We continue by presenting an algorithm for Fock boson sampling whose average-case time complexity is much faster when \( m \) is proportional to \( n \). In particular, when \( m = n \) our algorithm runs in approximately \( O(n \cdot 1.69^n) \) time on average. This result further increases the problem size needed to establish quantum computational advantage via the Fock scheme of boson sampling.

1. Introduction

The search for quantum computational advantage has garnered a great deal of interest and investment in recent years. One of the most promising candidates for this goal was introduced at STOC ’11 by Aaronson and Arkhipov (2013) who described an experimental set-up in linear optics that they termed boson sampling—more specifically Fock-state boson sampling. The experiment involves counting multiple indistinguishable photons entering and leaving a linear interferometer. For this set-up the counts at output modes can be shown to follow a distribution that is increasingly difficult to simulate on a classical computer as the problem size grows.

From the perspective of quantum computing, the experiment exhibits a calculation based on quantum effects of such a scale that it would be hard if not impossible to reproduce classically and therefore indicates quantum computational advantage (formerly termed ‘quantum supremacy’). Experimental Fock-state boson sampling has been demonstrated for various problem sizes (see e.g. Spring et al 2013, Bentivegna et al 2015, Broome et al 2013, Tillmann et al 2013, Crespi et al 2013, Spagnolo et al 2014, Latmiral et al 2016). In a Fock-state boson sampling experiment, \( n \) corresponds to the number of input photons and \( m \) the number of output modes. Increasing either of these is difficult in practice. Progress has therefore been slow (see Lund et al 2017, and the references therein) with the experimental record until recently being \( n = 5 \), \( m = 9 \) Wang et al 2016. In Wang et al 2019 demonstrated a Fock-state boson sampling experiment with \( m = 60 \) output modes and \( n = 20 \) input photons of which 14 are detected, following the model of Aaronson and Brod (2016).
The photonic experiment is designed to sample the output distribution of photons passing through a lossless linear interferometer. The input state

$$|\psi_{in}\rangle = |1_1, \ldots, 1_n, 0_{n+1}, \ldots, 0_m\rangle,$$

describes \( n \) indistinguishable photons simultaneously entering the first \( n \) modes of the interferometer. The interferometer is characterized by an \( m \)-dimensional Haar random unitary matrix \( A \) that determines the linear relation between the input and output creation and annihilation operators. The output measurement is a multiset of size \( n \) with elements drawn from the set of output modes \([m]\). It can be represented equivalently either by a non-decreasing array \( z = (z_1, \ldots, z_n) \) of elements of \([m]\) or by an array of multiplicities \( s = (s_1, \ldots, s_m) \) where \( s_j \) is the number of output photons in mode \( j, j = 1, \ldots, m \). The set of values of \( z \) is denoted by \( \Phi_{m,n} \) and the set of values of \( s \) by \( \Phi_{n,m}^s \). Each has cardinality \( (m+n-1)\vdash n \)—see Feller (1968), for example.

From Aaronson and Arkhipov (2013) or Gard et al (2015) and Schell (2004) with the \( z \) formulation above, the probability mass function (pmf) of a measurement outcome \( z \) can be shown to be

$$q(z|A) = \frac{1}{\mu(z)}|\text{Per} A^{[n]}|_z^2 \equiv \frac{1}{\mu(z)}\left| \sum_{\sigma} a_{\pi,\sigma} \right|^2, \quad z \in \Phi_{m,n},$$

where \( \text{Per} A^{[n]} \) is the permanent of \( A^{[n]} \) and in the definition the summation is for all \( \sigma \in \pi[n] \), the set of permutations of \([n]\). The \( n \times n \) matrix \( A^{[n]} \) is formed from \( A^{[1]} \), the first \( n \) columns of the given \( m \)-dimensional unitary matrix, so that row \( k \) of matrix \( A^{[n]} \) is row \( z_k \) in \( A^{[n]} \) for \( k = 1, \ldots, n \) and finally \( \mu(z) = \prod_{j=1}^m s_j! \). The repeated rows in \( A^{[n]} \) correspond to ‘collisions’ when multiple photons are in the same output mode.

For given \( A \), the Fock-state boson sampling problem is to simulate random samples independently and repeatedly from the pmf \( q(z|A) \) either with a quantum photonic device or with classical computing hardware. Aaronson and Arkhipov (2013) show that exact boson sampling is not efficiently solvable by a classical computer unless \( \text{P}^{#P} = \text{BPP}^{\text{NP}} \) and the polynomial hierarchy collapses to the third level. The lack of classical algorithms in this area supported the conclusion that the problem is hard. Initially the only known method was brute force evaluation of the probabilities of each multiset, as a preliminary to random sampling, requiring the calculation of \( \left( \begin{array}{c} m + n - 1 \\ n \end{array} \right) \) permanents of \( n \times n \) matrices, each one of which takes \( \mathcal{O}(n2^n) \) time with the fastest known algorithm. Subsequently we gave a faster classical sampling algorithm running in \( \mathcal{O}(n2^n \cdot \text{poly}(m,n)) \) time and linear space (Clifford and Clifford 2018). This suggested that at least 50 photons would be needed in any boson sampling experiment to demonstrate so-called quantum computational advantage. A more recent fine grained complexity theoretic analysis of boson sampling suggests that in fact 90 input photons will be needed to achieve quantum computational advantage (Dalzell et al 2018).

In practical terms increasing the number of input photons is not the only difficulty when performing boson sampling. Experiments also get increasingly difficult to perform as the number of output modes increases. Current experiments, for example, have typically set the number of modes to be a small multiple of the number of photons and it is likely this trend will continue in the near future.

The original hardness result for exact boson sampling of Aaronson and Arkhipov required that \( m \geq 2n \) as a crucial step in their reduction. For approximate boson sampling the number of outputs has to increase at least quadratically with the number of input photons for any known hardness results to apply. On the other hand, there is no existing evidence that approximate boson sampling is easier than exact boson sampling for any set of parameters. Moreover, Grier and Schaeffer (2016) showed that computing the permanent of a unitary matrix itself is \#P-hard. By applying this result to the proof technique of Aaronson and Arkhipov it follows that even with the number of output modes equal to the number of input photons, boson sampling cannot be performed in polynomial time classically unless \( \text{P}^{#P} = \text{BPP}^{\text{NP}} \). The result is consistent with the recent theoretical analysis by Bouland et al (2023) indicating that the calculation is hard, although practical consequences will depend critically on the exponential base when, for example, computation cost grows exponentially.

What then is the cost of classical calculation when the number of output modes is a small multiple of, or even equal to, the number of input modes? We show that if \( m = \theta n \) for constant \( \theta \geq 1 \) then it is indeed possible to sample from the boson sampling distribution significantly more quickly than was known before. The growth is still exponential but with a reduction in the exponential base. We establish the expected complexity averaged over all Haar-random realisations of the unitary \( A \). For simplicity, we first give the result for the case \( m = n \). The time complexity for the general case is given in section 6.

**Theorem 1.** The expected time complexity of Fock-state boson sampling with \( m = n \) is

$$\mathcal{O}(np^n),$$

where \( p = \frac{27}{16} \approx 1.69 \).

The additional space complexity on top of that needed to store the input is \( \mathcal{O}(m) \).
Our sampling algorithm also applies directly to the closely related problem of scattershot boson sampling Bentivegna et al 2015. From a mathematical perspective this produces no additional complications beyond the specification of a different $m \times n$ matrix. Once the matrix is specified we can apply our new sampling algorithm directly.

Straightforward enumeration shows that when $A$ is Haar-random the expected probability of a collision-free outcome is $\binom{n}{k}/\binom{m+n}{n}$. As a basis for their complexity argument, Aaronson and Arkhipov (2013) show that $m$ needs to grow faster than $n^2$ to ensure that collisions are rare. Similar calculations show that the expected number of singleton output modes is given by $mn(m-1)/[(m+n-1)(m+n-2)]$ which grows like $n^2/(1 + \theta)^2$ for large $n$ when $m = \theta n$. In particular, only a quarter of the output modes are singletons when $m = n$.

2. Related work

Terhal and DiVincenzo (2004) were the first to recognise that studying the complexity of sampling from low-depth quantum circuits could be useful for demonstrating a separation between quantum and classical computation. Since that time the goal of finding complexity separations for quantum sampling problems has received a great deal of interest. We refer the interested reader to Lund et al (2017), Harrow and Montanaro (2017) and Hangleiter and Eisert (2023) for surveys on the topic.

The search for faster classical algorithms for boson sampling has been of central interest since the problem was first explicitly formulated by Aaronson and Arkhipov. The first significant breakthrough was a Metropolised Independence Sampling (MIS) procedure utilised by Neville et al (2017). The overall approach of MIS (Liu 2001) is to build a Markov chain with the target distribution as its equilibrium by successively sampling from some easy to compute proposal distribution with acceptance probabilities depending on the relative likelihoods. Neville et al were able to provide numerical evidence that for limited problem sizes only approximately 200 such permanent computations were needed to take one sample approximately from the boson sampling distribution. The MIS approach is however necessarily approximate and does not give provable bounds on the quality of its approximation.

In a previous paper (Clifford and Clifford 2018) we presented a faster and provably correct boson sampling algorithm running in $O(n2^n + poly(m, n))$ time per sample, costing approximately two permanent calculations of $n \times n$ matrices for all values of $n$. The algorithm can also be extended and adapted to examine the effect of non-uniform losses and binned input/output modes. See Moylett et al (2019), Shchesnovich (2019), Brod and Oszmaniec (2020) and the references therein. The efficient calculation of permanents is key. Morse et al (2024) develop high performance implementations in relation to boson sampling.

Other approaches to complexity separation are based on Gaussian boson sampling (Lund et al 2014, Hamilton et al 2017, Kruse et al 2019) where input modes are prepared in squeezed states. The framework allows experimentation at a larger scale in terms of mode and photon count, providing strong support for quantum advantage. Zhong et al (2021) implement a 144-node system yielding up to 113 photon detection events, Madsen et al (2022) report experiments with 216 squeezed modes with an average of 125 photons and Deng et al (2023) report experiments with up to 255 photon clicks in the output using pseudo-photon-number-resolving detectors. In an alternative setting Young et al (2023) demonstrate boson sampling in an atomic system involving up to 180 atoms distributed among ~1000 sites in an optical lattice.

Classical algorithms for Gaussian boson sampling have been developed, building on those for the Fock protocol (Quesada and Arrazola 2020, Bulmer et al 2022, Quesada et al 2022). Oh (2024) proposes approximate Gaussian boson sampling algorithms based on tensor networks. Open source software libraries are available for various classical boson sampling computations, for example: Clifford et al (2017), Killoran et al (2019), Kolarovszki et al (2024).

3. Methods

3.1. Computing the permanent of low rank matrices

The complexity of computing the permanent of an arbitrary $k \times k$ complex matrix was shown to be $O(k^{2k})$ (Ryser 1963) and subsequently $O(k^2)$ (Nijenhuis and Wilf 1978, Glynn 2010). The computation time is decreased when there are several identical rows or columns, resulting in a matrix of reduced rank (Tichy 2011, Shchesnovich 2013, 2019). We will show how these ideas can be implemented in practice to produce a faster algorithm that suits our needs more closely.

Let $A^{(k)}$ be the first $k$ columns of a complex $m \times m$ matrix, $A$. Assume the rows of $A^{(k)}$ are distinct and let $B$ be a $k \times k$ matrix with repeated rows drawn from $A^{(k)}$. Specifically, let $z$ be a multiset of size $k$ with elements from
and let $s = (s_1, \ldots, s_m)$ be its associated array of multiplicities, then the $i$th row of $B$ will be row $z_i$ of $A^{[k]}$, for $i = 1, \ldots, k$.

According to Ryser’s formula

$$\text{Per}B = (-1)^k \sum_{T \subseteq [k]} (-1)^{|T|} \prod_{j \in T} b_{ij}. \quad (2)$$

The subset $T$ can be written as $\cup_{\nu=1}^m T_\nu$ where $T_\nu = \{i: z_i = \nu\}$. Note there are $s_\nu$ elements in $\{i: z_i = \nu\}$. It follows there are $s_\nu^r$ ways of choosing a subset $T_\nu$ of a given size $r_\nu$, when $r_\nu$ is between 0 and $s_\nu$. Furthermore $b_{ij} = a_{\nu,j}$ whenever $i \in T_\nu$.

Ryser’s formula can then be expressed as

$$\text{Per}B = (-1)^k \sum_{r_0=0}^{s_0} \cdots \sum_{r_m=0}^{s_m} (-1)^{n_1 + \cdots + n_m} \prod_{\nu=1}^m \left( s_\nu \right) \prod_{j=1}^k \left( \sum_{\nu=1}^m r_\nu a_{\nu,j} \right). \quad (3)$$

A straightforward implementation of the formula requires us to iterate over the tuples $(r_1, \ldots, r_m)$. Since each term in the summation requires $O(km)$ operations, the run time is then $O(km \prod_{\nu=1}^m (s_\nu + 1))$ as shown by Shchesnovich (2013). Similar complexity is achieved but with an improved constant factor overhead when starting from Glynn’s formula (Chin and Huh 2018).

To make this iteration as fast as possible we would ideally like to perform the iteration in such a way that at each stage we change only one of the $r_i$ values and these are changed by $\pm 1$. This can be achieved using Guan codes (otherwise known as generalised Gray codes) (Guan 1998), borrowing from the idea of Nijenhuis and Wilf (1978) who used a basic Gray code to speed up Ryser’s algorithm.

**Theorem 2.** Using Guan codes, $\text{Per}B$ can be calculated in $O(k \prod_{\nu=1}^m (s_\nu + 1))$ time.

**Proof.** There are $\prod_{\nu=1}^m (s_\nu + 1)$ terms in the outside set of summations in (3). By using Guan’s algorithm we can move through the tuples $(r_1, \ldots, r_m)$ exhaustively, adding or subtracting 1 from a single element. This means that the set of values $\sum_{\nu=1}^m s_\nu a_{\nu,j}$, $j \in [k]$ can be updated in $k$ operations. The product over $j \in [k]$ is a further $k$ operations. Updating the relevant binomial term is an $O(1)$ operation. The total operation count is then $O(k \prod_{\nu=1}^m (s_\nu + 1))$. □

### 3.2. Laplace expansion

We make extensive use of the Laplace expansion for permanents (see Marcus and Minc 1965, page 578), namely that for any $k \times k$ matrix $B = (b_{ij})$,

$$\text{Per}B = \sum_{\ell = 1}^k b_{i,\ell} \text{Per}B_{i,\ell},$$

where $B_{i,\ell}$ is the submatrix with row $i$ and column $\ell$ removed. Note that $B_{i,\ell}$ only depends on $B_{i,\ell}$, the submatrix of $B$ with the $k$-th row removed. An important consequence is that when $B$ is modified by changing its $k$-th row, the modified permanent can be calculated in $O(k)$ steps, provided the values $\{\text{Per}B_{i,\ell}^\nu\}$ are available. As explained in Clifford and Clifford (2018), we can take advantage of this observation to quickly compute a set of permanents of matrices each with one row differing from the other.

We now show that computation of all of the values $\{\text{Per}B_{i,\ell}^\nu, \ell \in [k]\}$ has the same time complexity as computing $\text{Per}B$, the permanent of a single $k \times k$ matrix when there are repeated rows in $B$. We derive this result for matrices with repeated rows using a combination of Ryser’s algorithm and Guan codes.

**Lemma 1.** Let $B$ be a $k \times k$ complex matrix with repeated rows specified by multiplicities $s$ and let $\{B_{i,\ell}^\nu\}$ be submatrices of $B$ with row $k$ and column $\ell$ removed, $\ell \in [k]$. The collection $\{\text{Per}B_{i,\ell}^\nu, \ell \in [k]\}$ can be evaluated jointly with the same time complexity as that of $\text{Per}B$, with $O(k)$ additional space.

**Proof of Lemma.** By applying Ryser’s formula (3) to $B_{i,\ell}^\nu$, for a given value of $\ell$ we have:

$$\text{Per}B_{i,\ell}^\nu = (-1)^k \sum_{r_0=0}^{s_0} \cdots \sum_{r_m=0}^{s_m} (-1)^{n_1 + \cdots + n_m} \prod_{\nu=1}^m \left( s_\nu \right) \prod_{j \in [k] \setminus \ell} w_j(\nu). \quad (4)$$

where $s_\nu$ is the multiplicity of repeated rows in $B_{i,\ell}^\nu$ and $w_j(\nu) = \sum_{\nu=1}^m s_\nu a_{\nu,j}$.

Working through values of $\nu$ in Guan code order, the terms $\{w_j(\nu), j \in [k]\}$ can be evaluated in $O(k)$ combined time for every new $\nu$. This is because successive $\nu$ arrays differ by one in a single element. The product
of the \(w_j(r)\) terms can be computed in \(O(k)\) time giving \(O(k \prod_{j=1}^{m} (s^j + 1))\) time to compute \(\text{Per}B^{\Omega}_k\) for a single value of \(\ell\), but of course this has to be replicated \(k\) times to cover all values of \(\ell\).

To compute \(\{\text{Per}B^{\Omega}_{k, \ell}, \ell \in [k]\}\) more efficiently we observe that each product \(\prod_{j \in [k] \setminus \ell} w_j(r)\) can be expressed as \(f_\ell b_\ell\) where \(f_\ell = \prod_{j=1}^{\ell-1} w_j(r), \ell = 2, \ldots, k\) and \(b_\ell = \prod_{j=\ell+1}^{m} w_j(r), \ell = 1, \ldots, k - 1\) are forward and backward cumulative products, with \(f_1 = b_k = 1\).

We can therefore compute all of the partial products \(\prod_{j \in [k] \setminus \ell} w_j(r)\) in \(O(k)\) time, giving an overall total time complexity of \(O(k \prod_{j=1}^{m} (s^j + 1))\) for jointly computing \(\{\text{Per} B^{\Omega}_{k, \ell}, \ell \in [k]\}\), and since \(s^j \leq s\), the time complexity is as claimed. Furthermore the computation time has constant factor overheads similar to that of computing \(\text{Per} B\). Other than the original matrix, space used is dominated by the two arrays of cumulative products, both of length \(O(k)\).

\[\square\]

4. Boson sampling algorithm

Clifford and Clifford (2018) provide the following algorithm for boson sampling:

**Algorithm 1.** Boson sampler: single sample \(z\) from \(q(z|A)\)

```plaintext
Require: \(m\) and \(n\) positive integers; \(m\)-dimensional Haar random unitary matrix, \(A\)
1: \(r \leftarrow \varnothing\) \quad \triangleright \text{EMPTY ARRAY}
2: \(A^{(0)} \leftarrow \text{Permute}(A^{(0)})\) \quad \triangleright \text{Randomly permute columns of } A^{(0)}
3: \(w_i \leftarrow [s_i]^2, i \in [m]\) \quad \triangleright \text{Make indexed weight array}
4: \(x \leftarrow \text{Sample}(w)\) \quad \triangleright \text{Sample index } x \text{ from } w
5: \(r \leftarrow (r, x)\) \quad \triangleright \text{Append } x \text{ to } r
6: \text{FOR } k \leftarrow 2 \text{ TO } n \text{ DO}
7: \(B_k^{(0)} \leftarrow A_k^{(0)}\)
8: \(\text{COMPUTE } \{\text{Per}B^{\Omega}_{k, \ell}, \ell \in [k]\}\) \quad \triangleright \text{AS LEMMA 1}
9: \(w_i \leftarrow [\sum_k \theta_i \text{Per}B^{\Omega}_{k, \ell} A_k^{(0)}, i \in [m]\) \quad \triangleright \text{USING LAPLACE EXPANSION}
10: \(x \leftarrow \text{Sample}(w)\)
11: \(r \leftarrow (r, x)\)
12: \text{END FOR}
13: \(z \leftarrow \text{INCSORT}(r)\) \quad \triangleright \text{SORT } r \text{ IN NON-DECREASING ORDER}
14: \text{RETURN } z
```

Calculation of the time complexity proceeds as in Clifford and Clifford (2018), with a few modifications to take account of repeated rows in evaluating permanents of the submatrices involved. At stage \(k\), let \(s^{(k)}\) be the multiplicities of the values in \((r_1, \ldots, r_k)\). The operation count in applying lemma 1 is \(O(k \prod_{j=1}^{m} (s^j + 1))\). Using the Laplace expansion for permanents, as described in section 3.2, an array of length \(m\) is obtained by summing \(k\) terms for each \(r_j \in [m]\). Taking a single sample from the pmf proportional to the array takes \(O(m)\) time. This gives the time complexity bound for stage \(k\) of \(O(k \prod_{j=1}^{m} (s^j + 1)) + O(mk)\) and hence a total operation count of

\[O\left(\sum_{k=1}^{n} k \prod_{j=1}^{m} (s^j + 1)\right) + O(mn^2).\]  

(5)

Importantly at the conclusion of stage \(k\), the \(k \times k\) matrix \(A^{(k)}_{r_1, \ldots, r_k}\) has been found. In other words at this intermediate stage a random submatrix has been drawn for the boson sampling problem with \(k\) photons and \(m\) output modes. Equivalently, the multiset formed by sorting \((r_1, \ldots, r_k)\) in non-decreasing order is then a random sample from the boson sampling distribution on \(\Psi_{n, k}\).

We now consider the average-case time complexity when the algorithm is applied with a random choice of Haar unitary matrix, \(A\).

5. Marginal uniformity of the boson sampling distribution

Arkhipov and Kuperberg have shown that the marginal boson sampling pmf averaged over Haar random unitaries is uniform on the space of multisets.
Theorem 3. With $q(z|A)$ as in (1) and $A$ a random matrix drawn from the $m$-dimensional Haar random unitary distribution, the marginal boson sampling pmf $Q(z)$ is given by

$$Q(z) = \mathbb{E}_q(z|A) = \left( \frac{m + n - 1}{n} \right)^{n-1}, \quad z \in \Phi_{m,n}.$$  

The proof is immediate from quantum theoretic considerations (Arkhipov and Kuperberg 2012). It can also be derived from properties of the Weingarten function in random matrix theory as follows.

Suppose $A$ is an $m$-dimensional Haar random unitary matrix. The Weingarten function is defined to be

$$Wg(\alpha, m) = \mathbb{E}(A_{i_1} \cdots A_{i_m} \bar{A}_{\alpha_{i_1}(1)} \cdots \bar{A}_{\alpha_{i_m}(n)}),$$

for $n \leq m$ and $\alpha$ in $\pi[n]$, where $\bar{A}_{\alpha}$ is the complex conjugate of $A_{\alpha}$ and $\pi[n]$ is the set of permutations of $[n]$.

We make use of the following result (Collins 2003, 2006).

Theorem 4. Let $i = (i_1, \ldots, i_n)$, $j = (j_1, \ldots, j_n)$ and $i' = (i'_1, \ldots, i'_n)$ be arrays of positive integers and let

$$\delta_i(i, i') = \prod_{k=1}^{n} \delta(i_k, i_{\alpha k}) \quad \text{for} \ \alpha \in \pi[n].$$

then

$$\mathbb{E}(A_{i_{i_1}} \cdots A_{i_{i_n}} \bar{A}_{\alpha_{i_1}(1)} \cdots \bar{A}_{\alpha_{i_n}(n)}) = \sum_{\alpha, \beta \in \pi[n]} \delta_i(i, i') \delta_j(j, j') Wg(\alpha \beta^{-1}, m).$$

As a immediate corollary we have: Corollary 1.

$$\sum_{\alpha \in \pi[n]} Wg(\alpha, m) = \frac{1}{m(m+1) \cdots (m+n-1)}.$$

Proof. Apply theorem 4 to $\mathbb{E}(|A_{i_{i_1}}|^{2n})$ with $i = i' = j = (1, \ldots, 1)$, so that

$$\mathbb{E}(|A_{i_{i_1}}|^{2n}) = \mathbb{E}(A_{i_1} \cdots A_{i_n} \bar{A}_{i_1} \cdots \bar{A}_{i_n}) = \sum_{\alpha, \beta \in \pi[n]} Wg(\alpha \beta^{-1}, m) = n! \sum_{\alpha \in \pi[n]} Wg(\alpha, m).$$

The last reduction follows since $\{\alpha \beta^{-1}; \ \alpha \in \pi[n]\} = \pi[n]$ for any given $\beta \in \pi[n]$. From (Petz and Réffy 2004), for example, with $W = |A_{i_{i_1}}|^2$ we have

$$\mathbb{E}(|A_{i_{i_1}}|^{2n}) = \mathbb{E}(W^n) = \int_0^1 (m - 1) w^n (1 - w)^{m-2} dw = \frac{n!}{m(m+1) \cdots (m+n-1)},$$

and the proof is complete. \hfill \Box

Turning now to the boson sampling distribution on multisets of size $n$ with elements in $[m]$ where a multiset is represented by an array $z = (z_1, \ldots, z_m)$ consisting of elements of $[m]$ in non-decreasing order. As before $\mu(z) = \prod_{j=1}^m s_j!$ where $s_j$ is the multiplicity of the value $j$ in $z$.

From the definition (1) we now have

$$\mu(z) Q(z) = \mathbb{E} \left[ \prod_{\sigma} \prod_{k=1}^{n} A_{s_{\sigma k}} \right], \quad \sigma \in \pi[n]$$

$$= \mathbb{E} \left[ \prod_{\sigma} \prod_{k=1}^{n} A_{s_{\sigma k}} \right] \left[ \prod_{\tau} \bar{A}_{s_{\tau k}} \right], \quad \tau \in \pi[n]$$

$$= \sum_{\sigma, \tau \in \pi[n]} \mathbb{E} \left[ \prod_{k=1}^{n} A_{s_{\sigma k}} \bar{A}_{s_{\tau k}} \right]$$

$$= \sum_{\sigma, \tau \in \pi[n]} \delta_\sigma(z, z) \delta_\tau(\sigma, \tau) \sum_{\alpha, \beta \in \pi[n]} \delta_i(i, i') Wg(\alpha \beta^{-1}, m)$$

$$= \sum_{\sigma, \tau \in \pi[n]} \delta_\sigma(z, z) Wg(\alpha \beta^{-1}, m) \sum_{\alpha, \beta \in \pi[n]} \delta_i(i, i')$$

(7)
using theorem 4. For each \( \beta \) the final summation in (7) is \( n! \), so the expectation becomes

\[
\mu(z) Q(z) = n! \sum_{\alpha, \beta \in \Pi [n]} \delta_{\alpha}(z, z) W \gamma (\alpha, \beta^{-1}, m)
\]

\[
= n! \sum_{\alpha \in \Pi [n]} \delta_{\alpha}(z, z) \sum_{\beta \in \Pi [n]} W \gamma (\alpha, \beta^{-1}, m)
\]

\[
= n! \sum_{\alpha \in \Pi [n]} \delta_{\alpha}(z, z) \frac{1}{m(m + 1) \cdots (m + n - 1)}
\]

\[
= \left( m + n - 1 \right)^{-1} \sum_{\alpha \in \Pi [n]} \delta_{\alpha}(z, z)
\]

from corollary 1 and collecting factorial terms The last term counts the number of ways that the elements of the array \( z \) can be permuted without changing the array, so that \( \sum_{\alpha \in \Pi [n]} \delta_{\alpha}(z, z) = \mu(z) \), and the result (6) follows.

6. Average-case time complexity of boson sampling

The average-case complexity of algorithm A is the expected value of (5) when \( A \) is drawn from the Haar random unitary distribution. We start by considering the term \( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \). Since the multiplicity array \( s^{(n)} \) is an alternative representation of the multiset \( z \), we know from theorem 3 that \( s^{(n)} \) is uniformly distributed over the set of all multiplicity arrays, \( \Phi_{m,n}^{*} \).

Recall that \( |\Phi_{m,n}^{*}| = \binom{m + n - 1}{m - 1} \) or equivalently \( \binom{m + n - 1}{n} \) as can be shown with the usual ‘stars and bars’ argument of Feller (1968), i.e. we place \( m - 1 \) bars at locations in \([1, \ldots, m + n - 1]\) and \( n \) stars at the remaining locations. The total number of such arrangements is \( \binom{m + n - 1}{m - 1} \). Adding further bars at each end, i.e. at locations 0 and \( m + n \), the elements of the associated array \( s^{(n)} \) are the counts of stars between bars.

Lemma 2. Suppose that \( s^{(n)} \) is sampled uniformly from \( \Phi_{m,n}^{*} \) the set of all possible multiplicity arrays then

\[
\text{E} \left( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \right) = \binom{m + n - 1}{n} \binom{2m + n - 1}{n}
\]

Proof. To see this, start with the stars and bars arrangement for a particular array \( s^{(n)} \) as above. Now consider adding a new bar between each existing neighbouring pair of bars. If there are \( s_{i}^{(n)} \) stars between a pair of bars, the new bar can be located in 1 of \( s_{i}^{(n)} + 1 \) places, for example if there is one star the new bar can be before or after it. The number of arrangements of stars and bars for a given array \( s^{(n)} \) is then \( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \).

Since \( s^{(n)} \) is uniformly distributed on \( \Phi_{m,n}^{*} \) and \( |\Phi_{m,n}^{*}| = \binom{m + n - 1}{n} \),

\[
\text{E} \left( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \right) = \binom{m + n - 1}{n} \sum_{s^{(n)} \in \Phi_{m,n}^{*}} \prod_{i=1}^{m} (s_{i}^{(n)} + 1).
\]

The result now follows because \( \sum_{s^{(n)} \in \Phi_{m,n}^{*}} \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \) is the total number of arrangements of the new and old stars and bars, i.e. the number of ways of placing \( 2m \) bars among \( 2m + n - 1 \) integer locations.

Corollary 2. With the conditions of lemma 2 and supposing that \( m = \theta n \) for some fixed value \( \theta \) then

\[
\text{E} \left( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \right) \sim \frac{\left( 2n + 1 \right)^{1/2}}{\left( 4\theta^{3}(\theta + 1)^{\theta+1} \right)^{n}}
\]

as \( n \to \infty \).

In particular \( \text{E} \left( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \right) \sim \left( 4/3 \right)^{1/2}(27/16)^{n} \approx 1.15(1.69)^{n} \) when \( \theta = 1 \), and \( \text{E} \left( \prod_{i=1}^{m} (s_{i}^{(n)} + 1) \right) \sim 2^{n} \) as \( \theta \to \infty \). The growth of the exponential base for intermediate values of \( \theta \) is illustrated in figure 1.

Proof. Apply Stirling’s formula to the factorial terms in (8).

Corollary 3. Let \( B \) a \( k \times k \) complex matrix with repeated rows having multiplicity \( s^{(k)} \) where \( s^{(k)} \) is uniformly distributed on \( \Phi_{m,k}^{*} \) then the expected running time of \( \text{PetB} \) is of order

\[
k \left( m + k - 1 \right) (2m + k - 1)^{1/k}
\]

Proof. This follows directly from lemma 2 and theorem 2.

We now prepare to prove the generalisation of theorem 1.
Theorem 5. The average-case time complexity of Fock-state boson sampling is bounded above by a term of order

\[ \frac{n(m+n)}{m} (m+n)^{-1} \left( \frac{2m+n}{n+1} \right) + n^2m \text{ as } m, n \to \infty. \]

In particular when \( m = \theta n \) for a fixed value of \( \theta \geq 1 \) this is of order \( \frac{(2\theta+1)^{2\theta+1}}{(4\theta^2(\theta+1)^{2\theta+1})} \) as \( n \to \infty. \)

Proof. Note that the first term in the time complexity of algorithm A given in (5) is equivalent to the total time complexity for evaluating the set of permanents of \( A_{rr}, r \in [n] \). In particular, evaluation of the permanent of the final matrix \( A_{n} \) has time complexity \( \Theta(n) \prod_{i=1}^{m} (s^{(\theta^*)}_{n+1} + 1) \) where \( s^{(\theta^*)} \) is the array of multiplicities in \( r. \) From lemma 2 this has a simple reduction in the average case, since \( s^{(\theta^*)} \) is uniformly distributed on \( \Phi^*_{m,n} \) by theorem 3.

At the intermediate stage \( k \), the matrix \( A^{[k]}_{r_{0},r_{1},...,r_{k}} \) can be viewed as the final matrix in a boson sampling algorithm with reduced size, i.e. where \( k \) columns are taken from \( A \) rather than \( n. \) Again the average-case complexity has a simple form from lemma 2 so the expected value of the first term in the time complexity of algorithm A given in (3), averaging over \( A, \) is

\[ \mathbb{E}\sum_{k=1}^{n} k \prod_{i=1}^{m} (s^{(\theta^*)}_{n+1} + 1) = \sum_{k=1}^{n} k \left( m + k - 1 \right)^{-1} \left( \frac{2m+k-1}{k} \right) \]

\[ = \frac{n(m+1)-m+1}{m+1} \left( \frac{m+n}{n+1} \right)^{-1} \left( \frac{2m+n}{n+1} \right) + \frac{2m(m-1)}{(m+1)(m+2)} \]

\[ \approx \Theta \left( \frac{n(m+n)}{m} \left( \frac{m+n}{n+1} \right)^{-1} \left( \frac{2m+n}{n+1} \right) \right) \text{ as } m, n \to \infty. \]

Incorporating the second term in (5) gives the complexity as claimed. Finally, Stirling’s formula gives the asymptotic form. \( \square \)

Acknowledgments

The authors would like to thank the anonymous referees who provided useful and detailed comments on an earlier version of the typescript.

Data availability statement

No new data were created or analysed in this study.

ORCID iDs

Raphaël Clifford ⓒ https://orcid.org/0000-0003-0599-0915
References

Aaronson S and Arkhipov A 2013 The computational complexity of linear optics Theory of Computing 9 143–252

Aaronson S and Brod J 2016 Bosonsampling with lost photons Phys. Rev. A 93 012335

Arkhipov A and Kuperberg G 2012 The bosonic birthday paradox Geometry & Topology Monographs 18 1–7

Bentivegna M et al 2015 Experimental scattered-boson boson sampling Science Advances 1 e1400255

Bouland A, Brod D, Datta I, Felleman B, Grier D, Hernandez F and Oszmaniec M 2023 Complexity-theoretic Foundations of Bosonsampling with a Linear Number of Modes arXiv:2312.00286

Brod D J and Oszmaniec M 2020 Classical simulation of linear optics subject to nonuniformal losses Quantum 4 267

Broome M A, Fedrizzi A, Rahimi-Keshari S, Dove J, Aaronson S, Ralph T C and White A G 2013 Photonic boson sampling in a tunable circuit Science 339 794–8

Bulmer J F et al 2022 The boundary for quantum advantage in Gaussian boson sampling Science advances 8 eabj9236

Chin S and Hub J 2018 Generalized concurrence in boson sampling Sci. Rep. 8 1–9

Clifford P and Clifford R 2017 CRAN R-package: Classical boson sampling https://CRAN.R-project.org/package=BosonSampling

Clifford P and Clifford R 2018 The classical complexity of boson sampling Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SIAM) 146–55

Collins B 2003 Moments and cumulants of polynomial random variables on unitary groups, the Itzykson-Zuber integral, and free probability International Mathematics Research Notices 2003 953–82

Collins B 2006 Integration with respect to the Haar measure on unitary, orthogonal and symplectic groups Commun. Math. Phys. 264 773–95

Crespi A, Oselelement R, Ramponi R, Brod D J, Galvão E F, Spagnolo N, Vitelli C, Maiorino E, Mataloni P and Sciarrino F 2013 Integrated multimode interferometers with arbitrary designs for photonic boson sampling Nat. Photonics 7 545–9

Dalzell A M, Harrow A W, Koh D E and La Placa R I. 2018 How many qubits are needed for quantum computational supremacy? arXiv:1805.05224

Deng Y-H et al 2023 Gaussian boson sampling with pseudo-photon-number-resolving detectors and quantum computational advantage Phys. Rev. Lett. 131 150601

Feller W 1968 Introduction to Probability Theory and its Applications 1 (Wiley) 1 Vol

Gard B T, Motes K R, Olson J P, Rohde P P and Dowling J P 2015 An introduction to boson-sampling From atomic to mesoscale: The role of quantum coherence in systems of various complexities (World Scientific) 167–92

Glynn D G 2010 The permanent of a square matrix Eur. J. Comb. 31 1887–91

Grier D and Schaferl R 2016 New Hardness Results for the Permanent using Linear Optics arXiv:1610.04670

Guan D-J 1998 Generalized Gray codes with applications Proceedings of the National Science Council. ROC(A) 22 841–8

Hamilton C S, Kruse R, Sansoni L, Barkhofen S, Silberhorn C and Jex I 2017 Gaussian boson Sampling Phys. Rev. Lett. 119 170501

Hangleiter D and Eisert J 2013 Computational advantage of Gaussian boson sampling New J. Phys. 11 95 035001

Harrow A W and Montanaro A 2017 Quantum computational supremacy Nature 549 203–9

Killoran N, Isaac J, Quesada N, Bergholm V, Amy M and Weedbrook C 2019 Strawberry fields: a software platform for photonic quantum computing Quantum 3 129

Kolarović Z. et al 2024 Piyavase: A Photonic Quantum Computer Simulation Software Platform arXiv:2307.00406

Krone R, Hamilton C S, Sansoni L, Barkhorn S, Silberhorn and C and Jex I 2018 Detailed study of Gaussian boson sampling Phys. Rev. A 100 032326

Latmiral L, Spagnolo N and Sciarrino F 2016 Towards quantum supremacy with lossy scattered-boson boson sampling New J. Phys. 18 11

Liu J 2001 Monte Carlo Strategies in Scientific Computing (Berlin: Springer) vol 75, 115

Lund A, Bremner M J and Ralph T C 2013 Boson sampling on a photonic chip Nat. Photonics 7 545–9

Lund A P, Laing A, Rahimi-Keshari S, Rudolph T, O’Brien J L and Ralph T C 2014 Boson sampling from a Gaussian State Phys. Rev. Lett. 113 100502

Madsen I S et al 2022 Quantum computational advantage with a programmable photonic processor Nature 606 75–81

Marcus M and Hic F 2015 Permanents The American Mathematical Monthly 72 577–91

Morse G, Rybotocki T, Kasapi A, Kolarović Z, Stojčić U, Koszic T, Menczer O, Oszmaniec M, Zimbardo Z and Rakyta P 2024 High performance boson sampling simulation via data flow engines New J. Phys. 26 033031

Moylett A E, García-Patrón R, Renema J J and Turner P S 2019 Classically simulating near-term partially-distinguishable and lossy boson sampling Quantum Science and Technology 5 015001

Neville A, Sparrow C, Clifford R, Johnston E, Birchall P M, Montanaro A and Laing A 2017 Classical boson sampling algorithms with superior performance to near-term experiments Nat. Phys. 13 1153–7

Nijenhuis A and Wilf H S 1978 Combinatorial Algorithms: For Computers and Calculators (Academic Press)

Oh C 2024 Classical algorithm for simulating experimental gaussian boson sampling Bulletin of the American Physical Society

Pérez D and Refiy J 2004 On asymptotics of large Haar distributed unitary matrices Period. Math. Hung. 109 13–17

Quesada N and Arrazola J M 2020 Exact simulation of Gaussian boson sampling in polynomial space and exponential time Phys. Rev. Research 2 023005

Quesada N et al 2022 Quadratic speed-up for simulating Gaussian boson sampling PRX Quantum 3 010306

Ryser H J 1963 Combinatorial mathematics, volume 14 of Carus Mathematical Monographs (American Mathematical Soc.)

Schellekens J 2007 Determinants: the classical algebraic approach (World Scientific)

Shchesnovich V 2013 On the asymptotic evaluation of bosonic probability amplitudes in linear unitary networks in the case of large number of bosons International Journal of Quantum Information 11 1350045

Shchesnovich V 2019 On the classical asymptotics of sampling from quantum interference of indistinguishable bosons arXiv:1904.02013v3

Spagnolo N et al 2014 Experimental validation of photonic boson sampling Nat. Photonics 8 615–20

Spring I B et al 2013 Boson sampling on a photonic chip Science 339 798–801

Terhal B M and DiVincenzo D P 2004 Adaptive quantum computation, constant depth quantum circuits and Arthur–Merlin games Quantum Inf. Comput. 4 134–45

Tichy M C 2011 Entanglement and interference of identical particles PhD thesis Freiburg University

Tillmann M, Dakić B, Heilmann R, Nolte S, Szameit A and Walther P 2013 Experimental boson sampling Nat. Photonics 7 540–4

Wang H et al 2016 Multi-photon boson-sampling machines beating early classical computers arXiv:1612.06956

Wang H et al 2019 Boson sampling with 20 input photons in 60-mode interferometers at 1014 state spaces arXiv:1910.09930

Young A W, Geller S, Eckner W J, Chirone N, Glancy S, Knill E and Kaufman M D 2012 An Atomic Boson Sampler arXiv:2307.06936

Zhong H et al 2021 Phase-programmable Gaussian boson sampling using stimulated squeezed light Phys. Rev. Lett. 127 180502