entropy

MDPI

Article

Classical Modelling of a Bosonic Sampler with Photon Collisions

Mikhail Umanskii and Alexey N. Rubtsov *

Russian Quantum Center, Skolkovo, Moscow 143025, Russia
* Correspondence: ar@rqc.ru

Abstract: The original formulation of the boson sampling problem assumed that little or no photon collisions occur. However, modern experimental realizations rely on setups where collisions are quite common, i.e., the number of photons M injected into the circuit is close to the number of detectors N. Here we present a classical algorithm that simulates a bosonic sampler: it calculates the probability of a given photon distribution at the interferometer outputs for a given distribution at the inputs. This algorithm is most effective in cases with multiple photon collisions, and in those cases, it outperforms known algorithms.

Keywords: boson sampling; photon collisions; computational complexity

check for updates

Citation: Umanskii, M.; Rubtsov, A.N. Classical Modelling of a Bosonic Sampler with Photon Collisions. *Entropy* **2023**, *25*, 210. https://doi.org/10.3390/e25020210

Academic Editor: Giuliano Benenti

Received: 5 December 2022
Revised: 17 January 2023
Accepted: 18 January 2023
Published: 21 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Quantum computers are computational devices that operate using phenomena described by quantum mechanics. Therefore, they can carry out operations that are not available for classical computers. Practical tasks are known which can be solved exponentially faster using quantum computers rather than classical ones. For example, the problem of integer factorization, which underlies the widely used RSA cryptosystem, can be solved by classical computers only in an exponential number of operations, whereas the quantum Shor's algorithm [1] can solve it in a polynomial number of operations. Due to the technological challenges of manufacturing quantum computers, quantum supremacy (the ability of a quantum computational device to solve problems that are intractable for classical computers) in practice remains an open question.

Boson sampling [2] is a good candidate for demonstrating quantum supremacy. Basically, boson samplers are linear-optical devices containing a number of non-classical sources of indistinguishable photons, a multichannel interferometer mixing photons of different sources, and photon detectors at the output channels of the interferometer. A more specific set-up to be addressed in this paper deals with the single photon sources. In this case, there are exactly M single photons injected into some or all N inputs of the interferometer. Performing multiple measurements of the photon counts at the outputs, one characterizes experimentally the many-body quantum statistics after the interferometer [3], given an input state and the interferometer matrix.

Boson samplers are not universal quantum computers, that is they cannot perform arbitrary unitary rotations in a high-dimensional Hilbert space of a quantum system. Nevertheless, a simulation of a boson sampler with a classical computer requires a number of operations exponential in M. It was shown [4] that a classical complexity of the boson sampling matches the complexity of computing the permanent of a complex matrix. This means that the problem of a boson sampling is #P-hard [5] and there are no known classical algorithms that solve it in polynomial time. The best known exact algorithm for computing the permanent of a $n \times n$ matrix is the Ryser formula [6], which requires $O(n2^n)$ operations. The Clifford–Clifford algorithm [7] is known to solve the boson sampling problem in $O(M2^M + NM^2)$ operations. This makes large enough bosonic samples practically intractable with classical computational devices. Although boson sampling does not allow for

arbitrary quantum computations, there are still practical problems that can be solved with boson sampling: for example, molecular docking [8], calculating the vibronic spectrum of a molecule [9,10] as well as certain problems in graph theory [11]. Boson sampling is also useful for statistical modeling [12] and machine learning [13,14].

There are several variants of boson sampling that aim at improving the photon generation efficiency and increasing the scale of implementations. For example, Scattershot boson sampling uses many parametric down-conversion sources to improve the single photon generation rate. It has been implemented experimentally using a 13-mode integrated photonic chip and six PDC photon sources [15]. Another variant is the Gaussian boson sampling [2,16], which uses the Gaussian input states instead of single photons. The Gaussian input states are generated using PDC sources, and it allows a deterministic preparation of the non-classical input light sources. In this variant, the relative input photon phases can affect the sampling distribution. Experiments were carried out with $N = 12$ [17] and $N = 100$ [18]. The latter implementation uses the PPKTP crystals as PDC sources and employs an active phase-locking mechanism to ensure a coherent superposition.

Any experimental set-up, of course, differs from the idealized model considered in theoretical modeling. Bosonic samplers suffer from two fundamental types of imperfections. First, the parameters of a real device, such as the reflection coefficients of the beam splitters and the phase rotations, are never known exactly. Varying the interferometer parameters too much can change the sampling statistics drastically so that modeling of an ideal device no longer makes much sense. Assume now that we know the parameters of the experimental set-up with great accuracy. Then what makes the device non-ideal is primarily photon losses, that is, not all photons emitted at the inputs are detected in the output channels. These losses happen because of imperfections in photon preparation, absorption inside the interferometer and imperfect detectors. There are different ways of modeling losses, for example by introducing the extra beam splitters [19] or replacing the interferometer matrix by a combination of lossless linear optics transformations and the diagonal matrix that contains transmission coefficients that are less than one [20].

Imperfections in middle-sized systems make them, in general, easier to emulate with classical computers [21]. It was shown [22] that with the increase of losses in a system the complexity of the task decreases. When the number of photons M' that arrive at the outputs is less than $\sqrt{M}$, the problem of a boson sampling can be efficiently solved using classical computers. On the other hand, if the losses are low, the problem remains hard for classical computers [23].

Photon collisions are a particular phenomenon that is present in nearly any experimental realization but were disregarded in the proposal by Aaronson and Arkhipov [4]. Originally it was proposed that the number of the interferometer channels is roughly a square of the number of photons in the set-up, $N \geq M^2$. In this situation, all or most of the photons arrive at a separate channel, that is no or a few of the photon collisions occur. In the experimental realizations [18], however, $N \approx M$.

Generally, a large number of photon collisions makes the system easier to emulate. For example, one can consider the extreme case that all photons arrive at a single output channel – the probability of such an outcome can be estimated within a polynomial time. The effect of the photon collisions on the computational complexity of a boson sampling has been previously studied [24]. A measure called the Fock state concurrence sum was introduced and it was shown that the minimal algorithm runtime depends on this measure. There is an algorithm for the Gaussian boson sampling that also takes advantage of photon collisions [25].

In this paper, we present the algorithm aimed at the simulation of the bosonic samplers with photon collisions. In the $N \approx M$ regime, our scheme outperforms the Clifford–Clifford method. For example, we consider an output state of the sampler with $M = N$ where one-half of the outputs are empty, and the other half is populated with 2 photons in each channel. Computing the probability of such an outcome requires us $O\left(N^2 3^{N/2}\right)$ operations. The speedup in states that have more collisions is even greater.

2. Problem Specification

Consider a linear-optics interferometer with N inputs and N outputs which is described by a given unitary $N \times N$ matrix U:

$$b_i^\dagger = \sum_{j=1}^{N} u_{ij} a_j^\dagger,\ a_i^\dagger = \sum_{j=1}^{N} u_{ji}^* b_j^\dagger, \tag{1}$$

where $a_i^\dagger$ and $b_i^\dagger$ are the creation operators acting on inputs and outputs, respectively. We will denote the input state as

$$|k\rangle = |k_1, k_2, \ldots, k_N\rangle = \prod_{i=1}^{N} \frac{1}{\sqrt{k_i!}} (a_i^\dagger)^{k_i} |0, 0, \ldots, 0\rangle, \tag{2}$$

where k_i is the number of photons in the i-th input. The output state will be denoted as

$$|l\rangle = |l_1, l_2, \ldots, l_N\rangle = \prod_{i=1}^{N} \frac{1}{\sqrt{l_i!}} (b_i^\dagger)^{l_i} |0, 0, \ldots, 0\rangle.$$

It follows from (1) and (2) that a specific input state corresponds to a set of output states that are observed with different probabilities:

$$|k_1, k_2, \ldots, k_N\rangle = \prod_{i=1}^{N} \frac{1}{\sqrt{k_i!}} (a_i^\dagger)^{k_i} |0, 0, \ldots, 0\rangle = \prod_{i=1}^{N} \frac{1}{\sqrt{k_i!}} \left(\sum_{j=1}^{N} u_{ji}^* b_j^\dagger \right)^{k_i} |0, 0, \ldots, 0\rangle$$

The product $\prod_{i=1}^{N} \left(\sum_{j=1}^{N} u_{ji}^* b_j^\dagger \right)^{k_i}$ can be written as

$$\prod_{i=1}^{N} \left(\sum_{j=1}^{N} u_{ji}^* b_j^\dagger \right)^{k_i} = (u_{11}^* b_1^\dagger + u_{21}^* b_2^\dagger + \ldots + u_{N1}^* b_N^\dagger)^{k_1} \cdot \ldots \cdot (u_{1N}^* b_1^\dagger + u_{2N}^* b_2^\dagger + \ldots + u_{NN}^* b_N^\dagger)^{k_N}. \tag{3}$$

After expanding, this expression will be a sum of terms that have the following form:

$$\alpha(l_1, l_2, \ldots, l_N) \prod_{i=1}^{N} \frac{1}{\sqrt{k_i!}} \left(b_i^\dagger \right)^{l_i} |0, 0, \ldots, 0\rangle = \alpha(l_1, l_2, \ldots, l_N) \prod_{i=1}^{N} \sqrt{\frac{l_i!}{k_i!}} |l_1, l_2, \ldots, l_N\rangle,$$

where $\alpha(l_1, l_2, \ldots, l_N)$ is a complex number that consists of the elements of U that correspond to the given output state. Therefore, the probability of observing an output state $|l_1, l_2, \ldots, l_N\rangle$ will be

$$|\langle l_1, l_2, \ldots, l_N k_1, k_2, \ldots, k_N\rangle|^2 = \left| \alpha(l_1, l_2, \ldots, l_N) \prod_{i=1}^{N} \sqrt{\frac{l_i!}{k_i!}} \right|^2 = |\alpha(l_1, l_2, \ldots, l_N)|^2 \prod_{i=1}^{N} \frac{l_i!}{k_i!}.$$

The problem is to determine the probabilities of all of the output states. The main difficulty lies in calculating the number $\alpha(l_1, l_2, \ldots, l_N)$ for the given input and output states. This paper presents an algorithm that solves this problem using the properties of the Fourier transform.

3. Algorithm Description

Let us define a function

$$g(t; \{Q_i\}) = \prod_{p=1}^{N} \left(\sum_{q=1}^{N} e^{i2\pi Q_q t}\ u_{qp}^* \right)^{k_p}, \tag{4}$$

where $\{Q_i\}$ is some fixed set of N natural numbers. The choice of $\{Q_i\}$ will later be discussed in detail. This function represents the expression (3), where the creation operators $b_j^\dagger$ are replaced with exponents $e^{i2\pi Q_j t}$ that oscillate with frequencies Q_j.

After expanding the expression (4), we get

$$g(t;\{Q_i\}) = \prod_{p=1}^{N}\left(\sum_{q=1}^{N} e^{i2\pi Q_q t}\, u_{qp}^*\right)^{k_p}$$

$$= (e^{i2\pi Q_1 t}u_{11}^* + e^{i2\pi Q_2 t}u_{21}^* + \ldots + e^{i2\pi Q_N t}u_{N1}^*)^{k_1} \cdot \ldots \cdot (e^{i2\pi Q_1 t}u_{1N}^* + e^{i2\pi Q_2 t}u_{2N}^* + \ldots + e^{i2\pi Q_N t}u_{NN}^*)^{k_N}$$

$$= \sum e^{i2\pi \sum_{i=1}^{N} l_i Q_i t}\, \alpha(l_1, l_2, \ldots, l_N),$$

where the sum is computed over all sets $\{l_1, \ldots, l_N\}$ such that $\sum_{i=1}^{N} l_i = M$.

Therefore, for each possible output state $|l\rangle = |l_1, l_2, \ldots, l_N\rangle$ there is the harmonic in $g(t;\{Q_i\})$ that has a frequency of $f(|l\rangle;\{Q_i\}) = \sum_{i=1}^{N} l_i Q_i$ and an amplitude of $\alpha(l_1, l_2, \ldots, l_N)$. The set of numbers $\{Q_i\}$ can be chosen in such a way that the harmonics do not overlap, i.e., there are no two outputs states $|l\rangle$ and $|l'\rangle$ with equal frequencies $f(|l\rangle;\{Q_i\}) = f(|l'\rangle;\{Q_i\})$.

If no harmonics overlap, then any of the numbers $\alpha(l_1, l_2, \ldots, l_N)$ can be found from the Fourier transform of the function $g(t;\{Q_i\})$. On the other hand, to calculate the probability of a specific state $|l\rangle$ it is sufficient to choose $\{Q_i\}$ in such a way that the frequency $f(|l\rangle;\{Q_i\})$ is unique in the spectrum, i.e., the frequency of any other state $|l'\rangle$ differs from the frequency of the state in consideration: $f(|l\rangle;\{Q_i\}) \neq f(|l'\rangle;\{Q_i\})\ \forall |l'\rangle \neq |l\rangle$.

An example of $g(t,\{Q_i\})$ with non-overlapping harmonics and its spectrum can be seen in Figure 1 (the choice of $\{Q_i\}$ used here is described in Section 3.1).

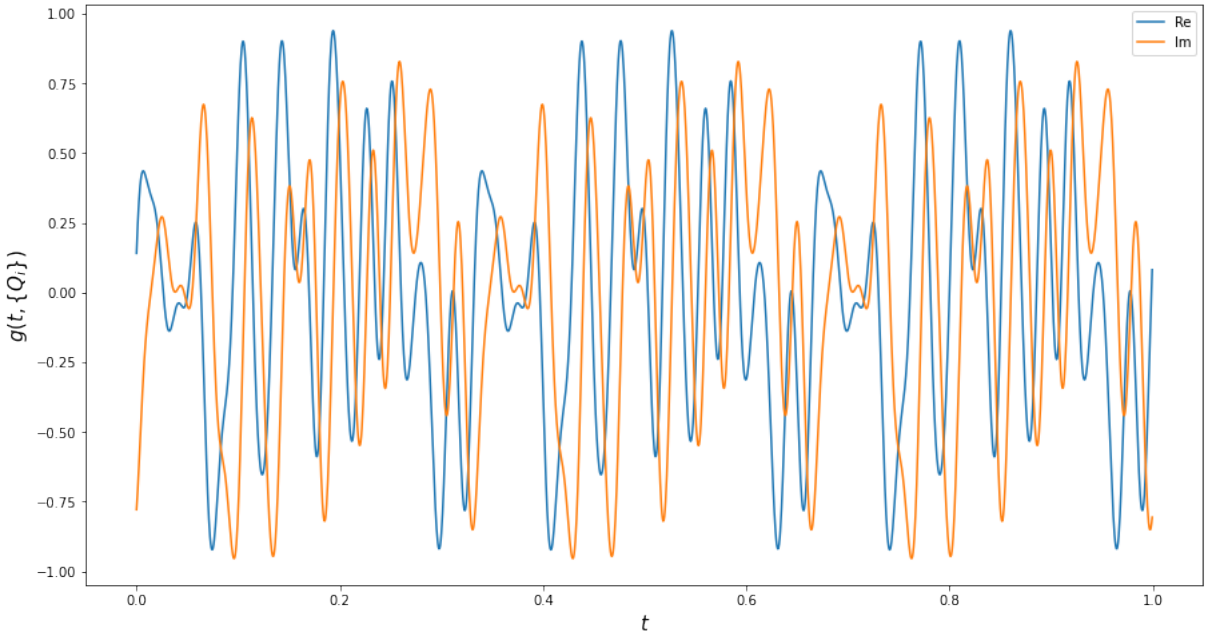


Figure 1. *Cont.*

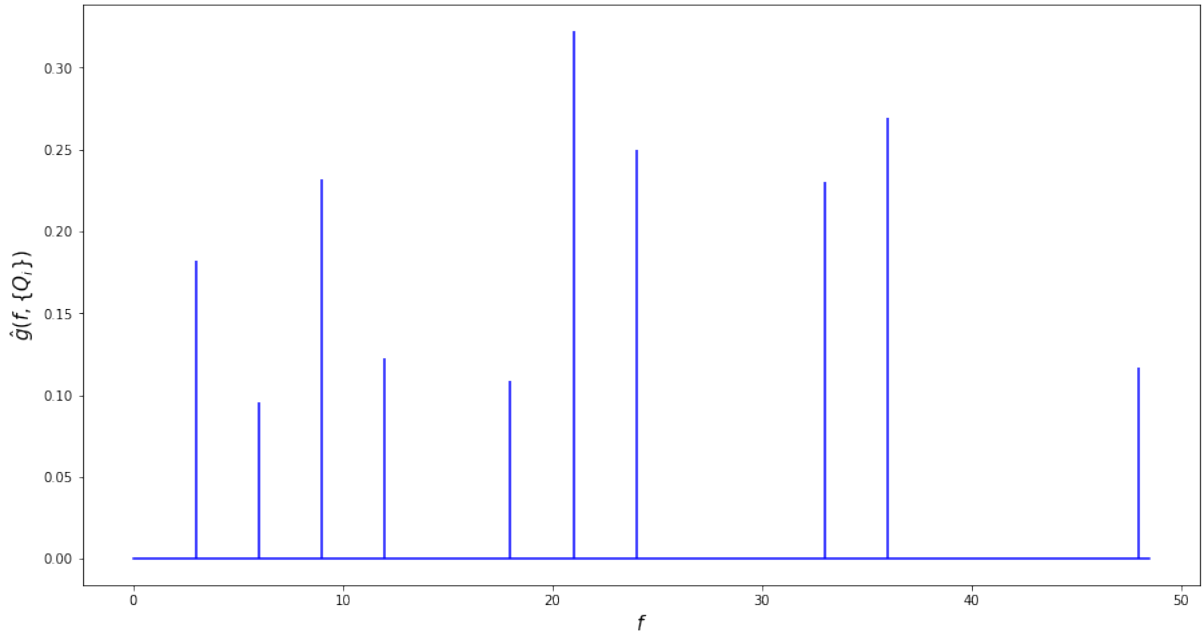


Figure 1. An example of the function $g(t, \{Q_i\})$ (the **top** picture) and its spectrum obtained as squared modulus of the Fourier transform of $g(t)$ (the **bottom** picture). A system with $N = M = 3$ is used, the input state is $|k\rangle = |1, 1, 1\rangle$. Each peak in the spectrum corresponds to one of ten possible output states.

3.1. The First Method of Choosing $\{Q_i\}$

Let us consider the methods of choosing $\{Q_i\}$ that will satisfy the necessary conditions on the spectrum. The first one consists of the following: let M be the total number of photons at the inputs, i.e., $M = \sum_{i=1}^{N} k_i$ for the input state $|k_1, k_2, \ldots, k_N\rangle$. We choose $Q = \{1, M+1, (M+1)^2, \ldots, (M+1)^{N-1}\}$, or $Q_i = (M+1)^{i-1}$. Then for an any output state $|l_1, l_2, \ldots, l_N\rangle$ the sum

$$\sum_{i=1}^{N} l_i Q_i = 1 \cdot l_1 + (M+1) \cdot l_2 + (M+1)^2 \cdot l_3 + \ldots + (M+1)^{N-1} \cdot l_N$$

will be a number that has a representation $\overline{l_N l_{N-1} \ldots l_1}$ in a positional numeral system with a radix $M+1$ (since $l_i < M+1 \ \forall i$). From the uniqueness of representation of numbers in positional numeral systems, it follows that every sum $\sum_{i=1}^{N} l_i Q_i$ (some number in a positional numeral system with a radix $M+1$) will correspond to exactly one set of numbers $l_1, l_2, \ldots, l_N$ (its representation in this numeral system; l_i being its digits).

Using this method of choosing $\{Q_i\}$ guarantees that the probability of any output state can be calculated from the spectrum of $g(t; \{Q_i\})$, since the frequencies $f(|l\rangle; \{Q_i\})$, $f(|l'\rangle; \{Q_i\})$ are different for any two output states $|l\rangle \neq |l'\rangle$.

3.2. The Second Method of Choosing $\{Q_i\}$

Another method of choosing $\{Q_i\}$ is useful when the goal is to compute the probability of one specific output state $|l\rangle$ when the input state $|k\rangle$ is given. This method does not guarantee that the frequencies will be different for any two output states, but it guarantees that the frequency of the state in consideration (the target frequency) $f(|l\rangle; \{Q_i\})$ will be unique in the spectrum. Note that in this case $Q_i = Q_i(|l\rangle)$, i.e., the choice of $\{Q_i\}$ depends on the output state.

This method of choosing $\{Q_i\}$ can be described in the following way:

$$Q_i = \begin{cases} \prod_{j=1}^{i-1}(l_j+1), \ l_i \neq 0 \\ 0, \ l_i = 0; \end{cases} \tag{5}$$

$$Q_1 = \begin{cases} 1, \ l_1 \neq 0 \\ 0, \ l_1 = 0. \end{cases}$$

Therefore if all of the outputs in the state $|l\rangle$ contain the photons, then $Q_1 = 1$, $Q_2 = (l_1+1)$, $Q_3 = (l_2+1)(l_1+1)$ and so on: Q_{i+1} is (l_i+1) times greater than Q_i.

Let us show that this method will actually lead to the target frequency being unique in the spectrum. Let $\{Q_i\} = \{Q_i(|l\rangle)\}$ be the frequencies calculated using the method described above. We need to prove that for any output state $|l'\rangle \neq |l\rangle$ it is true that $f(|l\rangle;\{Q_i\}) \neq f(|l'\rangle;\{Q_i\})$, i.e.,

$$\sum_{i=1}^{N} l_i Q_i \neq \sum_{i=1}^{N} l'_i Q_i.$$

Firstly, let us suppose that some of the outputs in the state $|l_1, l_2, \ldots, l_N\rangle$ contain 0 photons. Let $h_1, \ldots, h_K$ be the indices of the outputs that contain at least one photon: $l_{h_i} > 0 \ \forall i \in \{1, 2, \ldots, K\}$; $K < N$. Then the condition $f(|l\rangle;\{Q_i\}) \neq f(|l'\rangle;\{Q_i\})$ becomes

$$\sum_{i=1}^{K} l_{h_i} Q_{h_i} \neq \sum_{i=1}^{K} l'_{h_i} Q_{h_i},$$

since all the terms corresponding to empty outputs are zero in both sums ($Q_i = 0$ if the i-th output contains 0 photons).

Note that we can view it as a "reduced" system with the K outputs, in which the output state $|l\rangle$ contains at least one photon in every output. However, this system has one difference. Previously we considered the possible output states to be all the states that satisfy $\sum_{i=1}^{N} l'_i = M$ and $l'_i \geq 0 \ \forall i$. Now, in this "reduced" system we must consider all the output states such that $\sum_{i=1}^{K} l'_i \leq M$ and $l'_i \geq 0 \ \forall i$. This happens because the output states $|l'\rangle$ can have a non-zero amount of the photons in the outputs that were empty in $|l\rangle$; such outputs will have no effect on the frequency and they remain outside the "reduced" system.

Therefore, instead of a system where some outputs can be empty and some Q_i can be zero, but $\sum_{i=1}^{N} l_i = \sum_{i=1}^{N} l'_i$, we can consider the system where $l_i > 0 \ \forall i$ but $\sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i$. In this system $Q_i = \prod_{j=1}^{i-1}(l_j+1)$, and $Q_1 = 1$.

To prove the correctness of the algorithm, we must prove the following statement:

Theorem 1. *Let N be some natural number. Let $l_1, l_2, \ldots, l_N, l'_1, l'_2, \ldots l'_N$ be natural numbers that satisfy the following conditions:*
(1) $l_i > 0$, $l'_i \geq 0 \ \forall i$;
(2) $\sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i$;
(3) $\sum_{i=1}^{N} l_i Q_i = \sum_{i=1}^{N} l'_i Q_i$, where $Q_i = \prod_{j=1}^{i-1}(l_j+1)$ (and $Q_1 = 1$).
Then $l_i = l'_i \ \forall i$.

The proof of this statement can be found in Appendix A.

4. Parameters of the Fourier Transform

To calculate the Fourier transform of the function $g(t;\{Q_i\})$ we will use a fast Fourier transform (FFT). Firstly, we will define its parameters: the sampling interval Δt (or the sampling frequency $f_s = \frac{1}{\Delta t}$) and the number of data points K. The function will be calculated at points $n\Delta t$, $1 \leq n \leq K$. Since all the frequencies in the spectrum of $g(t;\{Q_i\})$

are natural numbers, they can be discerned with the frequency resolution of $\Delta f = 1$. The function therefore will be calculated in the points within an interval $[0; 1]$ which contains at least one period of each harmonic.

The sampling frequency $f_s = \frac{1}{\Delta t}$ is often chosen according to the Nyquist-Shannon theorem: if the Nyquist frequency $f_N = \frac{f_s}{2} = \frac{1}{2\Delta t}$ is greater than the highest frequency in the spectrum f_{max}, then the function can be reconstructed from the spectrum and no aliasing occurs. Therefore, one way of choosing the sampling frequency is $f_s = 2f_{max}$. It can be used with both methods of choosing $\{Q_i\}$.

Since the function is calculated in the points within the interval $[0; 1]$, the number of data points K is equal to the sampling frequency f_s. Optimization of the algorithm requires lowering the sampling frequency as much as possible.

If the goal is to calculate the probability of one specific state $|l\rangle$ and the second method of choosing $\{Q_i\}$ is used, then the sampling frequency f_s can be chosen to be lower than $2f_{max}$. This will lead to aliasing: a peak with frequency f will be aliased by the peaks with frequencies $f + kf_s$, $k \in \mathbb{Z}$. To correctly calculate the probability of the output state from the spectrum computed this way, the spectrum must not contain frequencies that satisfy $f(|l'\rangle; \{Q_i\}) = f(|l\rangle; \{Q_i\}) + kf_s$, $k \in \mathbb{Z}$. Note that it will not be possible to reconstruct the function $g(t; \{Q_i\})$ from such a spectrum.

We will show that the sampling frequency f_s for calculating the probability of the output state $|l\rangle$ using the second method of choosing $\{Q_i\}$ can be chosen to be greater by one than the target frequency $f = f(|l\rangle; \{Q_i(|l\rangle)\})$:

$$f_s = f + 1 = \sum_{i=1}^{N} l_i Q_i + 1$$

To prove this statement, we must show that the spectrum of $g(t; \{Q_i\})$ will not contain the frequencies $f' = f(|l'\rangle; \{Q_i\})$ that satisfy $f' = f + kf_s$, $k \in \mathbb{Z}$. This is shown by a theorem that is analogous to Theorem 1 yet has a weaker condition: equation in condition (3) is taken modulo f_s.

Theorem 2. *Let N be some natural number. Let $l_1, l_2, \ldots, l_N, l'_1, l'_2, \ldots l'_N$ be natural numbers that satisfy the following conditions:*
(1) $l_i > 0$, $l'_i \geq 0$ $\forall i$;
(2) $\sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i$;
(3) $\sum_{i=1}^{N} l_i Q_i \equiv \sum_{i=1}^{N} l'_i Q_i \mod (\sum_{i=1}^{N} l_i Q_i + 1)$, where $Q_i = \prod_{j=1}^{i-1}(l_j + 1)$ (and $Q_1 = 1$).
Then $l_i = l'_i$ $\forall i$.

The proof of this statement can be found in Appendix A.

5. Complexity of the Algorithm

Let us consider the computational complexity of this algorithm. The complexity of a fast Fourier transform of a data array of K points is $O(K \log K)$. The total complexity of the algorithm consists of the complexity of calculating $g(t; \{Q_i\})$ in K points and the complexity of a fast Fourier transform.

Computing $g(t; \{Q_i\})$ in each point is done in $\propto N^2$ operations: the expression for $g(t; \{Q_i\})$ consists of at most N factors, each of which can be computed in N additions, N multiplications and N exponentiations. If some of the inputs are empty, there will be fewer factors in the expression, and the resulting complexity will be lower.

When the first method of choosing $\{Q_i\}$ is used, the number of data points K is proportional to the highest frequency in the spectrum of $g(t; \{Q_i\})$, since the sampling frequency is chosen using the Nyquist-Shannon theorem. The frequencies corresponding to the outputs states, in this case, are equal to $\sum_{i=1}^{N} l_i Q_i = \sum_{i=1}^{N} l_i (M+1)^{i-1}$. The highest

frequency then is $M(M+1)^{N-1}$ and corresponds to the state where the last output contains all the photons. The total complexity of calculating all the probabilities will then be

$$O\Big(N^2M(M+1)^{N-1}\Big) + O\Big(M(M+1)^{N-1}\log\Big(M(M+1)^{N-1}\Big)\Big)$$

$$= O\Big(N^2M^N + NM^N\log M\Big).$$

When the second method of choosing $\{Q_i\}$ is used, the number of data points K depends on the frequency of the output state in consideration. This frequency is highest when photons are spread over outputs evenly. For the system with $M = mN$, $m \in \mathbb{N}$ this corresponds to the state where each output contains m photons. In this case, the highest frequency is equal to

$$\sum_{i=1}^{N} m(m+1)^{i-1} = m\frac{(m+1)^{N-1}-1}{m} = (m+1)^{N-1} - 1.$$

Therefore, the sampling frequency and the required number of data points will be $(m+1)^{N-1}$. The complexity of the algorithm in the worst case will be

$$O\Big(N^2(m+1)^N + N(m+1)^N\log(m+1)\Big).$$

In most states, however, photons will not be spread evenly between outputs, and outputs with a high number of photons will lower the sampling frequency and the complexity for calculating the probability of the state. This means that the more photon collisions are in a state, the better this algorithm performs. Let us consider several specific cases.

1. $M = N$, the goal is to compute the output state that contains 2 photons in one half of the outputs and 0 photons in the other half. The frequency corresponding to such a state will be

$$\sum_{i=1}^{N/2} 2\cdot 3^{i-1} = 3^{N/2-1} - 1,$$

and the complexity of the algorithm will be equal to

$$O\left(\frac{N^2}{2}(3^{N/2-1}-1) + (3^{N/2-1}-1)\log(3^{N/2-1}-1)\right) = O\Big(N^2 3^{N/2} + N3^{N/2}\Big) = O\Big(N^2 3^{N/2}\Big).$$

For comparison, the complexity of the Clifford–Clifford algorithm (which is $O(M2^M + NM^2)$) in this case will be equal to $O(N2^N)$.

2. $M = N^2$, the goal is to compute the output state that contains $2N$ photons in one half of the outputs and 0 photons in the other half. The frequency corresponding to such state will be $\sum_{i=1}^{N/2} 2N\cdot(2N+1)^{i-1} = (2N+1)^{N/2-1} - 1$, and the algorithm complexity will be

$$O\Big(N^2(2N)^{N/2} + N(2N)^{N/2}\log N\Big) = O\Big(N^2\cdot(2N)^{N/2}\Big)$$

Again, the complexity of the Clifford–Clifford algorithm, in this case, will be equal to $O(N^2 2^{N^2})$.

Weighted Average Complexity

We can measure the weighted average computational complexity of the algorithm described above by computing $\sum_i p_i C_i$, where i numbers measurement outcomes (output states) for the bosonic sampler, p_i is the probability to observe the ith state, C_i is the complexity of calculating p_i (assuming the second method of choosing $\{Q_i\}$ is used), and the sum is calculated over all possible outcomes.

We have computed this weighted average complexity for the systems with varying N. We set $M = N$, and $|k\rangle = |1, 1, \ldots, 1\rangle$ as the input state. The interferometer matrices

for those systems were randomly generated unitary matrices. Figure 2 shows that the weighted average complexity of the algorithm is significantly lower than the maximum complexity of the algorithm and their ratio decreases as N increases.

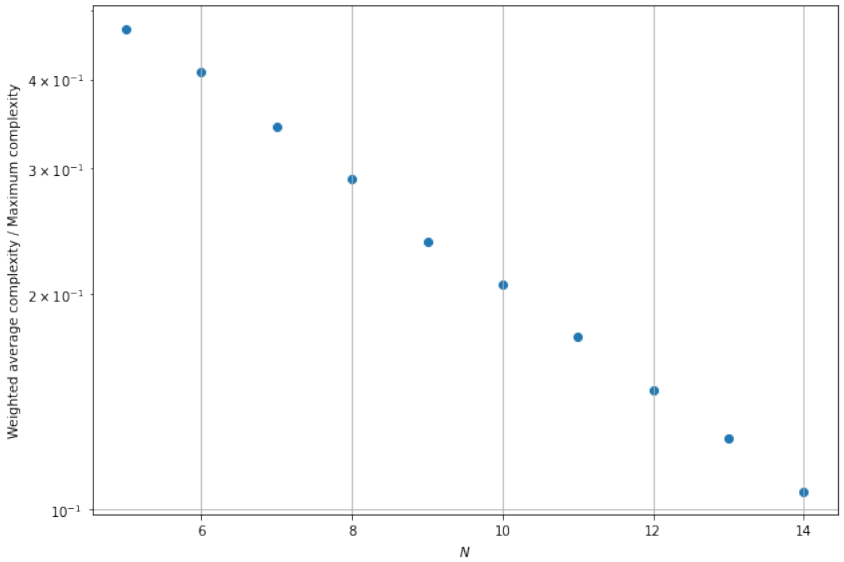


Figure 2. Decrease of the ratio of weighted average complexity to maximum complexity with the increase of N.

6. The Metropolis–Hastings

For systems with large N it might be computationally intractable to calculate the exact probability distribution of output states. The number of the possible output states scales with N and M as

$$C^{N-1}_{M+N-1} = \frac{(M+N-1)!}{(N-1)!(M+N-1-(N-1))!} = \frac{(M+N-1)!}{(N-1)!M!}.$$

Sampling from a probability distribution from which direct sampling is difficult can be done using the Metropolis–Hastings algorithm, which uses a Markov process. It allows us to generate a Markov chain in which points appear with frequencies that are equal to their probability. In our case, the points will be represented by the output states, i.e., sets of numbers $|l_1, l_2, \ldots, l_N\rangle$ such that $\sum_{i=1}^{N} l_i = M$.

We will require a transition function that will generate a candidate state from the last state in the chain. When the points are represented by real numbers, a candidate state can be chosen from a Gaussian distribution centered around the last point. In our case, however, the transition function will be more complex.

The transition function (Algorithm 1) must allow the chain to arrive in each of the possible states. It will be convenient to define it in the following way:

Algorithm 1 Transition function

$h := \{i : l_i > 0\}$ ▷ h is a set that contains indices of non-empty outputs
$K := |h|$ ▷ K is the amount of non-empty outputs
$r := random(\{1, \ldots, K\})$ ▷ We generate a random number $1 \leq r \leq K$
$l_{h_r} := l_{h_r} - 1$ ▷ Decrease the number of photons in h_r-th output
$s := random(\{1, \ldots, N\} \setminus h_r)$ ▷ We generate a random number $1 \leq s \leq N$ such that $s \neq r$
$l_s := l_s + 1$ ▷ Increase the number of photons in s-th output
return $|l_1, l_2, \ldots, l_N\rangle$

For the condition of a detailed balance to hold, we will require a function $p(l_1, \ldots, l_N; l'_1, \ldots, l'_N)$ which is equal to the probability of $|l'_1, \ldots, l'_N\rangle$ being the transition

function output when the last state in the chain is $|l_1, \ldots, l_N\rangle$. It is trivially constructed from the transition function.

Let u be the ratio of the exact probabilities of the states $|l'_1, \ldots, l'_N\rangle$ and $|l_1, \ldots, l_N\rangle$. The condition of a detailed balance will hold if the Markov chain will go from the state $|l_1, \ldots, l_N\rangle$ to the state $|l'_1, \ldots, l'_N\rangle$ with the probability

$$\alpha = u \cdot \frac{p(l'_1, \ldots, l'_N; l_1, \ldots, l_N)}{p(l_1, \ldots, l_N; l'_1, \ldots, l'_N)}.$$

Given the Markov chain, we can then calculate the approximate probability of a state by dividing the number of times this state occurs in the chain by the total number of steps of the chain.

Results

To demonstrate that the frequencies with which states appear in the Markov chain converge to the exact probability distribution, we have tested it on the system with $N = 10$, $M = 10$, $|k\rangle = |1, 1, \ldots, 1\rangle$ and a random unitary 10×10 matrix as the interferometer matrix. To calculate the distance between the exact and the approximate distribution we used cosine similarity:

$$S_C(p, q) = \frac{(\overrightarrow{p} \cdot \overrightarrow{q})}{|\overrightarrow{p}| \cdot |\overrightarrow{q}|} = \frac{\sum_i p_i q_i}{\sqrt{(\sum_i p_i^2)} \cdot \sqrt{(\sum_i q_i^2)}},$$

where p and q are some probability distributions. Namely, the value of $1 - S_C(p, q)$ is 0 when p and q are equal.

Figure 3 shows that $1 - S_C(p, q)$ decreases as the Markov chain makes more steps.

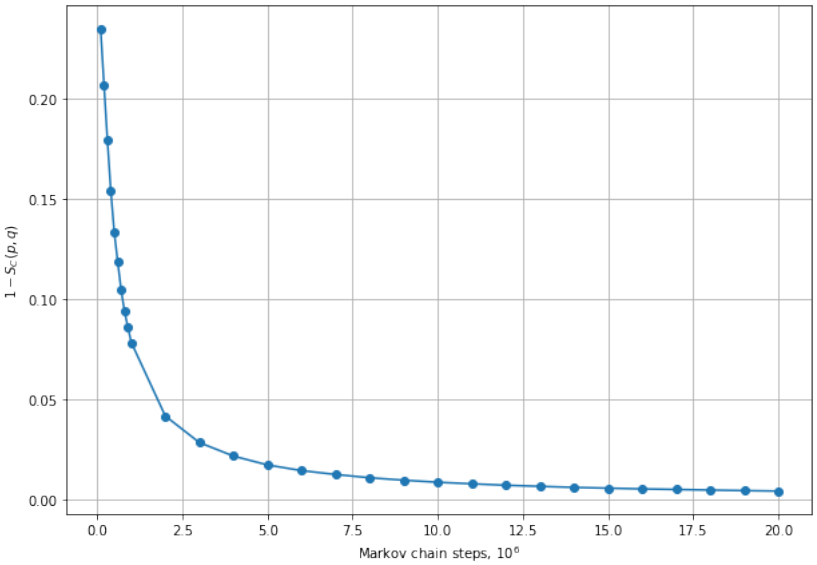


Figure 3. Convergence of the approximate probability distribution to the exact probability distribution.

7. Conclusions

We have presented a new algorithm for calculating the probabilities of the output states in the boson sampling problem. We have shown the correctness of the algorithm and calculated its computational complexity. This algorithm is simple in implementation as it relies heavily on the Fourier transform, which has numerous well-documented implementations.

The performance of this algorithm is better than the other algorithms in cases where there are many photon collisions. An example we give is an output state where all the

photons are spread equally across one-half of the outputs, with the other half of the outputs empty. In this case the algorithm requires $O\left(N^2 3^{N/2}\right)$ operations, while the Clifford–Clifford algorithm requires $O\left(N2^N\right)$ operations.

We have also proposed a method to approximately calculate the probability distribution in the boson sampling problem. It can be used when the system size is too large and calculating the exact probability distribution is intractable. Our results show that this algorithm indeed produces a probability distribution that converges to the exact probability distribution.

We plan to study further the application of the Metropolis–Hastings algorithm to approximate the boson sampling problem. When losses are modeled in the system, the probability distribution of the output states becomes concentrated. For example, when losses are high, the most probable states are those with many lost photons. When the losses are low, the probability is concentrated in the area where no or a few photons are lost. This property makes the Metropolis–Hastings algorithm especially effective in solving this problem.

Author Contributions: Conceptualization, A.N.R.; Software, M.U.; Formal analysis, M.U.; Investigation, M.U. All authors have read and agreed to the published version of the manuscript.

Funding: This research work was supported by the Russian Roadmap for the Development of Quantum Technologies, contract No. 868-1.3-15/15-2021.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Theorem A1. *Let N be some natural number. Let $l_1, l_2, \ldots, l_N, l'_1, l'_2, \ldots l'_N$ be natural numbers that satisfy the following conditions:*
(1) $l_i > 0$, $l'_i \geq 0$ $\forall i$;
(2) $\sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i$;
(3) $\sum_{i=1}^{N} l_i Q_i = \sum_{i=1}^{N} l'_i Q_i$, where $Q_i = \prod_{j=1}^{i-1}(l_j + 1)$ (and $Q_1 = 1$).
Then $l_i = l'_i$ $\forall i$.

Proof. We will prove this theorem by induction on N. The base case will be $N = 2$. Both the base case and the induction step will be proven by contradiction.

1. Base case.

Let us assume the opposite: $l_1 \neq l'_1$ and/or $l_2 \neq l'_2$.

The condition (3) will take the form

$$l_1 Q_1 + l_2 Q_2 = l'_1 Q_1 + l'_2 Q_2$$
$$l_1 + l_2(l_1 + 1) = l'_1 + l'_2(l_1 + 1).$$

We expand the brackets:

$$l_1 + l_2 + l_2 l_1 = l'_1 + l'_2 + l'_2 l_1.$$

According to condition (2), $l_1 + l_2 \geq l'_1 + l'_2$. Therefore,

$$l_2 l_1 \leq l'_2 l_1.$$

The condition (1) states that $l_1 > 0$. Therefore, $l_2 \leq l'_2$.

On the other hand, let us write condition (3) modulo $(l_1 + 1)$; the terms containing $(l_1 + 1)$ will be zero:

$$l_1 \equiv l'_1 \mod (l_1+1)$$

$l_1 \neq l'_1$, since otherwise it follows from condition (3) that $l_2(l_1+1) = l'_2(l_1+1) \implies l_2 = l'_2$ which leads to a contradiction (both $l_1 = l'_1$ and $l_2 = l'_2$). Therefore, since $l_1 \geq 0$ and $l'_1 \geq 0$, we have $l_2 > l'_2$. However, previously we have shown that $l_2 \leq l'_2$, which leads to a contradiction. This proves the base case.

2. Let us prove some general statements that will help us prove the induction step. Let us assume that the statement of the theorem is true for $N-1$. Then the following lemma holds for numbers $l_1, \ldots, l_N, l'_1, \ldots, l'_N$ that satisfy the conditions of the theorem for N: □

Lemma A1. *$\forall m \in \mathbb{N} : 1 \leq m \leq N$ the following is true: $l'_m = l_m - k_{m-1} + k_m(l_m+1)$, where k_m and k_{m-1} are natural numbers and $k_0 = k_N = 0$.*

Proof. We will prove this lemma by induction on m. First we prove the base case $m = 1$. Let us write down the expression from condition (3) of the theorem modulo (l_1+1) (all terms that contain (l_1+1) will turn to zero and only the first ones from each side will remain):

$$l_1 \equiv l'_1 \mod (l_1+1). \tag{A1}$$

Then

$$l'_1 = l_1 + k_1(l_1+1), \tag{A2}$$

where k_1 is an integer. Since $l'_1 \geq 0$, k_1 must be natural. Since $k_0 = 0$, the equation $l'_1 = l_1 - k_0 + k_1(l_1+1)$ is true.

Now let us assume that the statement of this lemma holds for all i such that $1 \leq i \leq m-1$. Let us write down the expression from condition (3) of the theorem modulo $(l_m+1)(l_{m-1}+1)\ldots(l_1+1)$—this will turn to zero all the terms, except for first m on both sides (note that Q_m has a form of $(l_{m-1}+1)(l_{m-2}+1)\ldots(l_1+1)$):

$$l_1Q_1 + l_2Q_2 + \ldots + l_mQ_m \equiv l'_1Q_1 + l'_2Q_2 + \ldots + l'_mQ_m \mod (l_m+1)(l_{m-1}+1)\ldots(l_1+1).$$

Then

$$l'_1Q_1 + l'_2Q_2 + \ldots + l'_mQ_m = l_1Q_1 + l_2Q_2 + \ldots + l_mQ_m + k_m(l_m+1)(l_{m-1}+1)\ldots(l_1+1), \tag{A3}$$

where k_m is an integer. Using $l'_1 = l_1 + k_1(l_1+1)$ and dividing by (l_1+1) , we get

$$k_1 + l'_2 + l'_3(l_2+1) + \ldots + l'_m(l_{m-1}+1)(l_{m-2}+1)\ldots(l_2+1) =$$

$$= l_2 + l_3(l_2+1) + \ldots + l_m(l_{m-1}+1)(l_{m-2}+1)\ldots(l_2+1) + k_m(l_m+1)(l_{m-1}+1)\ldots(l_2+1).$$

Applying sequentially $l'_i = l_i - k_{i-1} + k_i(l_i+1)$ and dividing by (l_i+1) for all i from 2 to $m-1$ we get

$$k_{m-1} + l'_m = l_m + k_m(l_m+1) \implies l'_m = l_m - k_{m-1} + k_m(l_m+1),$$

where $k_{m-1} \geq 0$ by the assumption of the induction step and $k_m \geq 0$ since $l'_m \geq 0$.

Let us specifically consider the case $m = N$. The expression (A3) will take the following form:

$$l_1Q_1 + l_2Q_2 + \ldots + l_NQ_N = l'_1Q_1 + l'_2Q_2 + \ldots + l'_NQ_N + k_N(l_N+1)(l_{N-1}+1)\ldots(l_1+1).$$

Using condition (3) of the theorem we get

$$k_N(l_N+1)(l_{N-1}+1)\dots(l_1+1)=0 \implies k_N=0,$$

which proves the lemma.

We now go back to the theorem. Suppose the theorem is false for N but true for $N-1$. Lemma 1 has some corollaries that are used in proving the induction step. Firstly, suppose $l_1=l'_1$ but $\exists i>1 : l_i \neq l'_i$. Then the condition (3) will take the form of

$$\begin{aligned} l_2(l_1+1)+l_3(l_2+1)(l_1+1)+\ldots+l_N(l_{N-1}+1)\dots(l_1+1)= \\ =l'_2(l_1+1)+l'_3(l_2+1)(l_1+1)+\ldots+l'_N(l_{N-1}+1)\dots(l_1+1), \end{aligned}$$

which can be divided by (l_1+1) to get

$$l_2+l_3(l_2+1)+\ldots+l_N(l_{N-1}+1)\dots(l_2+1)=l'_2+l'_3(l_2+1)+\ldots+l'_N(l_{N-1}+1)\dots(l_2+1).$$

Moreover, the condition (2) can be written as

$$\sum_{i=2}^{N} l_i \geq \sum_{i=2}^{N} l'_i.$$

It means that numbers $l_2, l_3, \dots, l_N, l'_2, l'_3, \dots l'_N$ satisfy the conditions of the theorem for $N-1$, which is assumed to be true. Therefore, $l_i=l'_i \ \forall i>1$ which is a contradiction. As a result, $l'_1 \neq l_1$; using (A2) we get $k_1>0$.

Secondly, let us write down condition (2) of the theorem with the equation $l'_i = l_i - k_{i-1} + k_i(l_i+1)$ (which is given by Lemma 1):

$$\sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i \implies \sum_{i=1}^{N}(l'_i-l_i)=\sum_{i=1}^{N}(-k_{i-1}+k_i(l_i+1))\leq 0.$$

After expanding it we get

$$\begin{aligned} \sum_{i=1}^{N}(-k_{i-1}+k_i(l_i+1))=\sum_{i=1}^{N}(-k_{i-1}+k_il_i+k_i))= \\ =-k_0+k_1l_1+k_1-k_1+k_2l_2+k_2-k_2+k_3l_3+k_3+\ldots-k_{N-1}+k_Nl_N+k_N= \\ =-k_0+k_1l_1+k_2l_2+\ldots+k_Nl_N+k_N\leq 0. \end{aligned}$$

Since $k_0=k_N=0$,

$$\sum_{i=1}^{N} k_il_i \leq 0.$$

However, the condition (1) states that $l_i>0 \ \forall i$, Lemma 1 states that $k_i \geq 0 \ \forall i>1$, and we have shown above that $k_1>0$. Therefore $k_il_i \geq 0 \ \forall i$ and $k_1l_1>0$, which means that $\sum_{i=1}^{N} k_il_i>0$. We get a contradiction which proves the theorem. □

Theorem A2. *Let N be some natural number. Let $l_1, l_2, \dots, l_N, l'_1, l'_2, \dots l'_N$ be natural numbers that satisfy the following conditions:*
(1) $l_i>0$, $l'_i \geq 0 \ \forall i$;
(2) $\sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i$;
(3) $\sum_{i=1}^{N} l_iQ_i \equiv \sum_{i=1}^{N} l'_iQ_i \mod (\sum_{i=1}^{N} l_iQ_i+1)$, where $Q_i=\prod_{j=1}^{i-1}(l_j+1)$ (and $Q_1=1$).
Then $l_i=l'_i \ \forall i$.

Proof. We will prove this theorem by contradiction. Suppose there are such numbers $l_1, l_2, \dots, l_N, l'_1, l'_2, \dots l'_N$ that satisfy the conditions of the theorem, but $\exists i : \ l_i \neq l'_i$. Let us write down condition (3) of the theorem in the following way:

$$\sum_{i=1}^{N} l'_i Q_i = \sum_{i=1}^{N} l_i Q_i + q(\sum_{i=1}^{N} l_i Q_i + 1),$$

where $q \in \mathbb{Z}$ and $q \geq 0$ since $\sum_{i=1}^{N} l'_i Q_i > 0$. If $q = 0$, than Theorem 1 can be applied and $l_i = l'_i \ \forall i$, which is a contradiction. From now on we will consider the case $q > 0$. We can rearrange the expression which is multiplied by q:

$$\begin{aligned}\sum_{i=1}^{N} l_i Q_i + 1 = 1 + l_1 + l_2(l_1+1) + \ldots + l_N(l_{N-1}+1)\ldots(l_1+1) = \\ = (l_1+1)(1 + l_2 + l_3(l_2+1) + \ldots + l_N(l_{N-1}+1)\ldots(l_2+1)) = \\ = (l_1+1)(l_2+1)(1 + l_3 + l_4(l_3+1) + \ldots + l_N(l_{N-1}+1)\ldots(l_3+1)) = \\ = \ldots = (l_1+1)(l_2+1)\ldots(l_N+1) = Q_{N+1}.\end{aligned}$$

Now condition (3) of the theorem can be rewritten as

$$\sum_{i=1}^{N} l'_i Q_i = \sum_{i=1}^{N} l_i Q_i + q Q_{N+1}.$$

Let us define $l'_{N+1} \equiv 0$, $l_{N+1} = q > 0$. Then

$$\sum_{i=1}^{N+1} l'_i Q_i = \sum_{i=1}^{N+1} l_i Q_i. \tag{A4}$$

The numbers $l_1, \ldots, l_{N+1}; l'_1, \ldots, l'_{N+1}$ satisfy the conditions of the Theorem 1. The condition (1) is satisfied because $l_{N+1} = q > 0$ and $l'_{N+1} \equiv 0$. Since

$$\sum_{i=1}^{N+1} l_i > \sum_{i=1}^{N} l_i \geq \sum_{i=1}^{N} l'_i = \sum_{i=1}^{N+1} l'_i,$$

The condition (2) of the Theorem 1 is also satisfied. The condition (3) is identical to the expression (A4). Therefore, $l_i = l'_i \ \forall i$; but

$$l_{N+1} = q \neq 0 = l'_{N+1}.$$

This is a contradiction which proves the theorem. □

References

1. Shor, P.W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **1997**, *26*, 1484–1509. [CrossRef]
2. Lund, A.; Laing, A.; Rahimi-Keshari, S.; Rudolph, T.; O'Brien, J.; Ralph, T. Boson Sampling from a Gaussian State. *Phys. Rev. Lett.* **2014**, *113*, 100502. [CrossRef] [PubMed]
3. Gard, B.T.; Motes, K.R.; Olson, J.P.; Rohde, P.P.; Dowling, J.P. An Introduction to Boson-Sampling. In *From Atomic to Mesoscale*; World Scientific: Singapore, 2015; pp. 167–192. [CrossRef]
4. Aaronson, S.; Arkhipov, A. The Computational Complexity of Linear Optics. *Arxiv* **2010**, arxiv: 1011.3245.
5. Aaronson, S. A linear-optical proof that the permanent is #P-hard. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **2011**, *467*, 3393–3405. [CrossRef]
6. Ryser, H.J. *Combinatorial Mathematics*; American Mathematical Society: Providence, RI, USA, 1963; Volume 14.
7. Clifford, P.; Clifford, R. The Classical Complexity of Boson Sampling. In Proceedings of the 2018 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), New Orleans, LA, USA, 7–10 January 2018.
8. Banchi, L.; Fingerhuth, M.; Babej, T.; Ing, C.; Arrazola, J.M. Molecular docking with Gaussian Boson Sampling. *Sci. Adv.* **2020**, *6*, eaax1950. [CrossRef] [PubMed]
9. Huh, J.; Guerreschi, G.G.; Peropadre, B.; McClean, J.R.; Aspuru-Guzik, A. Boson sampling for molecular vibronic spectra. *Nat. Photonics* **2015**, *9*, 615–620. [CrossRef]
10. Huh, J.; Yung, M.H. Vibronic Boson Sampling: Generalized Gaussian Boson Sampling for Molecular Vibronic Spectra at Finite Temperature. *Sci. Rep.* **2017**, *7*, 7462. [CrossRef]

11. Brádler, K.; Dallaire-Demers, P.L.; Rebentrost, P.; Su, D.; Weedbrook, C. Gaussian boson sampling for perfect matchings of arbitrary graphs. *Phys. Rev. A* **2018**, *98*, 032310. [CrossRef]
12. Jahangiri, S.; Arrazola, J.M.; Quesada, N.; Killoran, N. Point processes with Gaussian boson sampling. *Phys. Rev. E* **2020**, *101*, 022134. [CrossRef]
13. Schuld, M.; Brádler, K.; Israel, R.; Su, D.; Gupt, B. Measuring the similarity of graphs with a Gaussian boson sampler. *Phys. Rev. A* **2020**, *101*, 032314. [CrossRef]
14. Banchi, L.; Quesada, N.; Arrazola, J.M. Training Gaussian boson sampling distributions. *Phys. Rev. A* **2020**, *102*, 012417. [CrossRef]
15. Bentivegna, M.; Spagnolo, N.; Vitelli, C., Flamini, F.; Viggianiello, N.; Latmiral, L.; Mataloni, P.; Brod, D.J.; Galvão, E.F.; Crespi, A.; et al. Experimental scattershot boson sampling. *Sci. Adv.* **2015**, *1*, e1400255. [CrossRef] [PubMed]
16. Hamilton, C.S.; Kruse, R.; Sansoni, L.; Barkhofen, S.; Silberhorn, C.; Jex, I. Gaussian Boson Sampling. *Phys. Rev. Lett.* **2017**, *119*, 170501. [CrossRef] [PubMed]
17. Zhong, H.S.; Peng, L.C.; Li, Y.; Hu, Y.; Li, W.; Qin, J.; Wu, D.; Zhang, W.; Li, H.; Zhang, L.; et al. Experimental Gaussian Boson sampling. *Sci. Bull.* **2019**, *64*, 511–515. [CrossRef] [PubMed]
18. Zhong, H.S.; Wang, H.; Deng, Y.H.; Chen, M.C.; Peng, L.C.; Luo, Y.H.; Qin, J.; Wu, D.; Ding, X.; Hu, Y.; et al. Quantum computational advantage using photons. *Science* **2020**, *370*, 1460–1463. [CrossRef]
19. Oh, C.; Noh, K.; Fefferman, B.; Jiang, L. Classical simulation of lossy boson sampling using matrix product operators. *Phys. Rev. A* **2021**, *104*, 022407. [CrossRef]
20. García-Patrón, R.; Renema, J.J.; Shchesnovich, V. Simulating boson sampling in lossy architectures. *Quantum* **2019**, *3*, 169. [CrossRef]
21. Popova, A.S.; Rubtsov, A.N. Cracking the Quantum Advantage threshold for Gaussian Boson Sampling. *arXiv* **2021**, arXiv:2106.01445.
22. Qi, H.; Brod, D.J.; Quesada, N.; García-Patrón, R. Regimes of Classical Simulability for Noisy Gaussian Boson Sampling. *Phys. Rev. Lett.* **2020**, *124*, 100502. [CrossRef]
23. Aaronson, S.; Brod, D.J. BosonSampling with lost photons. *Phys. Rev. A* **2016**, *93*, 012335. [CrossRef]
24. Chin, S.; Huh, J. Generalized concurrence in boson sampling. *Sci. Rep.* **2018**, *8*, 6101. [CrossRef] [PubMed]
25. Bulmer, J.F.F.; Bell, B.A.; Chadwick, R.S.; Jones, A.E.; Moise, D.; Rigazzi, A.; Thorbecke, J.; Haus, U.U.; Vaerenbergh, T.V.; Patel, R.B.; et al. The boundary for quantum advantage in Gaussian boson sampling. *Sci. Adv.* **2022**, *8*, eabl9236. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.