Classical Algorithms for Measurement-Adaptive Gaussian Circuits

Changhun Oh[1,*] and Youngrong Lim[2]

[1] Department of Physics, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea
[2] Department of Physics, Chungbuk National University, Cheongju, Chungbuk 28644, South Korea

(Received 16 September 2025; revised 30 November 2025; accepted 6 January 2026; published 11 February 2026; corrected 14 October 2026)

Gaussian building blocks are essential for photonic quantum information processing, and universality can be practically achieved by equipping Gaussian circuits with adaptive measurement and feedforward. The number of adaptive steps then provides a natural parameter for computational power. Rather than assessing power only through sampling problems—the usual benchmark—we follow the ongoing shift toward tasks of practical relevance and study the quantum mean-value problem, i.e., estimating observable expectation values that underpin simulation and variational algorithms. More specifically, we analyze bosonic circuits with adaptivity and prove that when the number of adaptive measurements is small, the mean-value problem admits efficient classical algorithms even if a large amount of non-Gaussian resources are present in the input state, whereas less constrained regimes are computationally hard. This yields a task-level contrast with sampling, where non-Gaussian ingredients alone often induce hardness, and provides a clean complexity boundary parameterized by the number of adaptive measurement-and-feedforward steps between classical simulability and quantum advantage. Beyond the main result, we introduce classical techniques—including a generalization of Gurvits' second algorithm to arbitrary product inputs and Gaussian circuits—for computing the marginal quantities needed by our estimators, which may be of independent interest.

DOI: 10.1103/rpl7-ylg8

I. INTRODUCTION

Quantum computers are promised to provide computational power for solving various problems, such as integer factorization [1] and time-dynamics simulation of quantum systems [2], which are believed to be intractable for any classical computer. However, due to demanding experimental requirements, including a large number of qubits and small error rates, achieving quantum computational advantage for solving practical problems remains highly challenging. Thus, although the ultimate goal is to build a universal quantum computer, many recent experiments have focused on demonstrating quantum advantage through sampling problems that are relatively easier to implement in experiments [3–12] while many debates are going on regarding the precise boundary between classical simulability and genuine quantum advantage [13–26].

In bosonic systems, particularly in photonic platforms, the most prominent paradigm for quantum advantage is boson sampling due to a relatively more straightforward requirement, a linear-optical circuit [27–29]; consequently, numerous experiments have been conducted to demonstrate quantum advantage using boson sampling [7–12]. Despite its hardness results [27–29], it is also widely recognized that universal quantum computation requires more than linear-optical circuits, such as non-Gaussian operations or measurement-and-feedforward operations [30–34].

In theory, nonlinear gates such as Kerr interactions are sufficient for universality [35]. In practice, however, such gates are extremely difficult to realize because photons rarely interact, and inevitable loss and noise further degrade the effect [36]. Consequently, feasible bosonic gates in experiments are largely limited to Gaussian unitaries. Since nonlinearity remains indispensable for universality, many experiments instead rely on non-Gaussian measurements combined with postselection or feedforward [30,37–42]. For example, heralding can generate single-photon states from two-mode squeezed vacua, while Gottesman-Kitaev-Preskill states can be approximately prepared by postselecting certain outcomes in Gaussian boson-sampling circuits [41–43]. Alternatively, non-Gaussian resources can be supplied entirely through the input states, provided that the measurement and feedforward operations are available, for the universal

*Contact author: changhun0218@gmail.com

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

quantum computation [34]. This has motivated recent experimental efforts to go beyond the boson-sampling framework by incorporating non-Gaussian resources, such as measurement-induced nonlinear effects, to unlock more versatile and powerful photonic processors. Representative directions include nonlinear boson sampling [40] and adaptive boson sampling [44,45].

Importantly, some of these recent works aim not only to demonstrate sampling-based quantum advantage but also to move toward more practical computational advantages, ultimately aligned with building a universal quantum computer. From this perspective, it becomes increasingly critical to understand the complexity of simulating bosonic circuits for tasks beyond sampling. A particularly important computational task is the quantum mean-value problem because of its generality and practicality (e.g., Refs. [46,47]). More specifically, the quantum mean-value problem encompasses various practical problems, such as variational quantum algorithms (VQAs) and kernel-based machine learning applications [44,45,48]. Furthermore, the quantum mean-value problem for a universal quantum circuit is known to be BQP-complete [49,50]; hence, this problem essentially captures the universal quantum computer's power, provided that the necessary resources are supplied. This shift reflects a broader transition: extending the role of photonic devices from demonstrating quantum advantage in sampling problems to realizing practical advantages in tasks more relevant to real-world applications. In this context, clarifying the boundary between classical and quantum computational power is essential.

In this work, we investigate the complexity of simulating bosonic circuits with measurement and feedforward operations, focusing on how the computational cost of the simulation depends on the number of measurements for feedforward operations (see Table I for the main results). Among various notions of simulation [51,52], we mainly analyze and compare two: sampling and the quantum mean-value problem. While non-Gaussian resources generally render the sampling problem classically intractable [27–29,53,54], we show that the quantum mean-value problem can remain efficiently solvable classically even with highly non-Gaussian inputs, as long as the number of adaptive measurements is limited. Hence, our main results characterize the classical costs of these tasks as functions of the number of measurements and feedforward. Additionally, it suggests that the requirement for the hardness of the quantum mean-value problem is significantly more demanding. We show this by presenting classical algorithms that can efficiently estimate the expectation value of a product observable when (i) the number of measurements used for feedforward is at most constant and (ii) the Hilbert-Schmidt norm of the observable is at most polynomially large.

In this work, we use the term *measurement-adaptive Gaussian circuit* for the following class of photonic architectures. We consider a collection of bosonic modes that undergo layers of Gaussian unitaries, possibly interleaved with L layers of measurements on some of the modes. The measurement outcomes are processed classically and can be used to choose the subsequent Gaussian unitaries (feedforward). We allow general, possibly non-Gaussian, product input states on some modes (e.g., single-photon or GKP states), while the dynamical gates acting during the circuit are Gaussian. The special case $L = 0$ corresponds to a nonadaptive Gaussian circuit, whereas $L \geq 1$ yields a measurement-adaptive Gaussian circuit in our terminology.

More specifically, considering practical relevance, we analyze three circuit families in detail: (i) Gaussian circuits without feedforward, (ii) Gaussian circuits with photon-number-resolving detection for feedforward, and (iii) Gaussian circuits with Gaussian measurements for feedforward. It is worth emphasizing that if a sufficient number of photon-number measurements or Gaussian measurements for feedforward is provided, the circuits (ii) and (iii) become a universal quantum computer [30,34]. Hence, the quantum mean-value problem becomes BQP-complete. Therefore, understanding the complexity of the

TABLE I. Comparison of sampling and quantum mean-value estimation complexities for different families of bosonic quantum circuits: Gaussian circuit without feedforward, Gaussian circuits augmented by photon-number counting and Gaussian feedforward, and Gaussian circuits augmented by Gaussian measurement and Gaussian feedforward. Here, M is the number of modes and L is the number of measurement-and-adaptive operations. Asterisks represent the results from this work.

	Sampling (Gaussian input and measurement)	Quantum mean-value problem
Gaussian circuits without feedforward	Efficient [31–33]	*Efficient (Sec. V A, Theorem 1)
Gaussian circuits + photon-number counting	*Efficient when $L = O(\log M)$ (Sec. V B, Theorem 2)	*Efficient when $L = O(1)$ (Sec. V B, Theorem 3), BQP-complete for general L [30]
Gaussian circuits + Gaussian measurement	Efficient [31–33]	*Efficient when $L = O(1)$ (Sec. V C, Theorem 4), BQP-complete for general L [34]

quantum mean-value problem is crucial for delineating regimes where such circuits remain classically simulable (for example, with a bounded amount of adaptivity) and regimes where they are powerful enough to enable universal quantum computation. We also emphasize that our classical algorithms are efficient when the Hilbert-Schmidt norm of the observable is at most polynomially large, which is precisely the regime targeted by many proposals for *practical* quantum advantage. In photonic platforms, such proposals typically consider expectation values of few-mode or k-local observables, for example, local Hamiltonian terms, correlation functions, witnesses, or other figures of merit used as quantum-advantage benchmarks, whose Hilbert-Schmidt norms grow at most polynomially with the system size (e.g., photonic VQAs [55–60]). In these scenarios, additive-error estimation of such observables is the natural accuracy notion, so our results identify a regime where practically motivated mean-value tasks remain efficiently classically simulable, even though the same architectures become BQP-complete once unrestricted adaptivity is allowed.

To prove the efficiency of our classical algorithms, we exploit low-rank structures of matrices arising in these circuits, inspired by Gurvits' second algorithm, which enables us to compute the marginal probabilities of Fock-state boson sampling [27,61]. It is well known that low-rankness often significantly reduces the computational complexity in evaluating functions such as the permanent, hafnian, and loop hafnian [62–64]. Using these observations, we design efficient classical algorithms for simulating bosonic circuits with a limited number of measurements for feedforward. In particular, by extending Gurvits' second algorithm to more general setups, our algorithms exploit the low-rank structure of matrices associated with Gaussian circuits, which enables us to approximate relevant quantities with significantly reduced computational costs. We expect that our extension of Gurvits' second algorithm is of independent interest for the complexity analysis of bosonic systems.

Our paper is organized as follows. In Sec. II, we first compare our work to previously known results to clarify our contributions. In Sec. III, we provide preliminaries for our main results. In Sec. IV, we provide the details of the problem setups. In Sec. V, we provide the main results about classical algorithms that simulate measurement-adaptive Gaussian circuits. More specifically, in Sec. V A, we consider Gaussian circuits without feedforward and provide the classical algorithms that solve the sampling and quantum mean-value problems associated with the circuits, respectively. In Sec. V B, we introduce measurement-adaptive Gaussian circuits by photon-number-resolving detectors and provide the classical algorithms solving the sampling and quantum mean-value problems, respectively. In Sec. V C, we consider measurement-adaptive Gaussian circuits by Gaussian measurements and provide the classical algorithms solving the sampling and quantum mean-value problems, respectively. In Sec. VI, we provide the main techniques that are necessary for the classical algorithms. In Sec. VII, we conclude.

II. RELATED PREVIOUS STUDIES

Conceptually, our study aims to find an analogous classical algorithm in bosonic systems to those for simulating quantum circuits composed mainly of Clifford gates with a few T-gates [65,66]. The analogy is that, in qubit systems, circuits with stabilizer input states, Clifford operations, and computational basis measurements are efficiently simulable on classical computers by the Gottesman-Knill theorem [67], whereas the addition of T-gates provides the necessary "magic" for universal quantum computation. In bosonic systems, Gaussian input states, Gaussian circuits, and Gaussian measurements play the role of stabilizer-Clifford circuits, i.e., easy to classically simulate, while non-Gaussian gates correspond to T-gates, which make the circuit hard to classically simulate [31,33]. Moreover, just as T-gates can be simulated via gadgets that rely on measurement and feedforward, we also consider measurement-induced non-Gaussian elements in bosonic circuits [30,37].

This analogy has motivated a line of work on the classical simulation of Gaussian circuits augmented by limited non-Gaussian resources [53,54,68–74] while many of which particularly focus on strong simulation, i.e., computing the output (marginal) probabilities [53,54,69,72]. For example, Chabaud and Walschaers [53] and Mele *et al.* [74] analyzed the role of non-Gaussian resources in continuous-variable circuits and identified which types of resources are necessary to achieve quantum computational advantage. Chabaud *et al.* [69], Dias *et al.* [72], and Hahn *et al.* [54] developed classical algorithms for simulating Gaussian circuits with a small number of non-Gaussian gates or inputs, primarily targeting sampling problems and their strong simulations. Other works, such as Veitch *et al.* [68], explored the role of Wigner function negativity as a resource for quantum computational power, while Bourassa *et al.* [70] investigated fast classical simulation methods for restricted classes of photonic circuits. More recently, Calcluth *et al.* [73] provided sufficient conditions for efficient simulation of bosonic circuits under resource constraints. Overall, these studies mainly focus on classical simulation in the context of sampling problems, with a focus on strong simulations, rather than the quantum mean-value problem that we emphasize here.

Meanwhile, there is also a rich literature on estimating output probabilities of bosonic circuits with additive error, which is a special case of the quantum mean-value problem. In particular, quasi-probability-based methods have been studied for estimating output probability distributions

[51,75,76]. For example, Pashayan *et al.* [51,75] showed that such estimation techniques can be carried out classically with complexity depending on the negativity of the quasi-probability representation. While these methods have been applied to several interesting systems, they have not been specifically developed for the class of bosonic circuits we consider in this work. Additionally, whereas their algorithms concentrate on estimating output probabilities, our work addresses a more general problem: the quantum mean-value problem. Furthermore, our method does not explicitly depend on the negativity of the quantum circuits, which is in stark contrast to the previous works, making our method more generally applicable.

Finally, our previous work initiated the study of the complexity of the quantum mean-value problem in nonadaptive linear-optical circuits [77]. While the main motivation there was to investigate the practicality of boson-sampling-type architectures, passive linear optics alone is not believed to render a universal quantum computer. More generally, Gaussian circuits by themselves are also nonuniversal; universality is recovered only when one supplements Gaussian or linear-optical unitaries with suitable non-Gaussian resources or measurement-based feedforward, as in the Knill-Laflamme-Milburn (KLM) scheme and Gaussian cluster-state architectures. Motivated by this, in the present work, we extend this line of research from passive linear optics to a broader class of bosonic circuits, namely measurement-adaptive Gaussian circuits, where Gaussian unitaries are interleaved with a finite number of measurement-and-feedforward layers acting on possibly non-Gaussian resource states. We emphasize that there is increasing attention to the adaptive circuits due to their potential practical applications [44,45,78]. From a technical standpoint, we employ a similar method for the main routine of the classical algorithms. However, extending from linear-optical circuits to Gaussian circuits with measurement and feedforward requires a more challenging subroutine, which is computing marginal probabilities for general Gaussian circuits. To do this, we extend Gurvits' second algorithm, which works only for Fock states and linear-optical circuits, to arbitrary product states and Gaussian circuits [27,61]. To this end, we also provide a classical algorithm for computing the overlap of a low-mode quantum state evolved through a linear-optical circuit with an arbitrary product state.

III. PRELIMINARY

In this section, we provide some preliminaries for the main results (more details can be found in Refs. [79–81]). We will mainly focus on M-mode bosonic systems throughout this work. Here, the annihilation and creation operators for ith mode are denoted by $\hat{a}_i$ and $\hat{a}_i^\dagger$, respectively, and they satisfy the canonical commutation relation, $[\hat{a}_i, \hat{a}_j^\dagger] = \delta_{ij}$ and $[\hat{a}_i, \hat{a}_j] = [\hat{a}_i^\dagger, \hat{a}_j^\dagger] = 0$. In such systems, any M-mode operator $\hat{O}$ can be written as a linear combination of displacement operators,

$$\hat{O} = \frac{1}{\pi^M} \int d^{2M}\boldsymbol{\alpha}\, \chi_{\hat{O}}(\boldsymbol{\alpha}) \hat{D}^\dagger(\boldsymbol{\alpha}), \tag{1}$$

where $\chi_{\hat{O}}(\boldsymbol{\alpha}) \equiv \mathrm{Tr}[\hat{D}(\boldsymbol{\alpha})\hat{O}]$ is called the characteristic function of an operator $\hat{O}$, and $\hat{D}(\boldsymbol{\alpha})$ is an M-mode displacement operator of an amplitude $\boldsymbol{\alpha} \in \mathbb{C}^M$ which is a tensor product of single-mode displacement operators, i.e., $\hat{D}(\boldsymbol{\alpha}) = \otimes_{i=1}^M \hat{D}(\alpha_i) = \otimes_{i=1}^M e^{\alpha_i \hat{a}_i^\dagger - \alpha_i^* \hat{a}_i}$ with $\alpha_i \in \mathbb{C}$. The displacement operator with zero amplitude is equal to the identity operator $\hat{D}(\mathbf{0}) = \hat{\mathbb{1}}_M$, where $\hat{\mathbb{1}}_M$ denotes the identity operator for the M-mode system. An important property of the displacement operator that we frequently use is the following twirling relation:

$$\frac{1}{\pi^M} \int d^{2M}\boldsymbol{\alpha}\, \hat{D}(\boldsymbol{\alpha}) \hat{O} \hat{D}^\dagger(\boldsymbol{\alpha}) = \mathrm{Tr}[\hat{O}] \hat{\mathbb{1}}_M. \tag{2}$$

Note that the convention of integration is $\int d^{2M}\boldsymbol{\alpha} = \int_{\mathbb{C}^M} \prod_{j=1}^M d^2\alpha_j$ and $\int_{\mathbb{C}} d^2\alpha_j = \int_{-\infty}^{\infty} d\,\mathrm{Re}\,\alpha_j\, d\,\mathrm{Im}\,\alpha_j$, where $\alpha_j = \mathrm{Re}\,\alpha_j + i\,\mathrm{Im}\,\alpha_j$ throughout this work.

A. Gaussian states, circuits, and measurements

In this work, we frequently consider Gaussian states, Gaussian unitary circuits, and Gaussian measurements [32,80,81]. Thus, we provide definitions for these objects.

First of all, Gaussian states are defined as the ones whose characteristic function (or equivalently, the Wigner function) is Gaussian. The states that are not Gaussian are called non-Gaussian states.

Gaussian unitary circuits are those that map Gaussian states to Gaussian states. Otherwise, the circuits are referred to as non-Gaussian. An essential property of Gaussian unitary circuits is that by the Bloch-Messiah decomposition, any Gaussian unitary circuit can be written as a linear-optical circuit $\hat{U}$, the M-mode tensor product of single-mode squeezing gates $\hat{S}(\boldsymbol{r}) = \otimes_{i=1}^M \hat{S}(r_i)$, and another linear-optical circuit $\hat{V}$, i.e., $\hat{G} = \hat{D}(\boldsymbol{\beta})\hat{U}\hat{S}(\boldsymbol{r})\hat{V}$, where $\boldsymbol{\beta} \in \mathbb{C}^M$ is the displacement vector and $\boldsymbol{r} \in \mathbb{R}^M$ is the squeezing parameter vector. Here, a linear-optical circuit, a restricted family of Gaussian circuits, is defined as an operator that transforms bosonic annihilation operators as

$$\hat{U}^\dagger \hat{a}_i \hat{U} = \sum_{j=1}^M U_{ij} \hat{a}_j, \tag{3}$$

where U is the associated $M \times M$ unitary matrix to the linear-optical circuit $\hat{U}$. Another important property that is frequently exploited is that a Gaussian unitary

circuit transforms a displacement operator into another displacement operator [32,80,81],

$$e^{2i\,\mathrm{Im}(\boldsymbol{\alpha}^\dagger\boldsymbol{\beta})}\hat{D}^\dagger(\boldsymbol{\alpha}') = \hat{G}^\dagger\hat{D}^\dagger(\boldsymbol{\alpha})\hat{G}, \tag{4}$$

where $\boldsymbol{\beta}$ is the displacement in $\hat{G}$'s Bloch-Messiah decomposition and $\boldsymbol{\alpha}' \in \mathbb{C}^M$ is a function of $\boldsymbol{\alpha}$ that depends on the Gaussian circuit $\hat{G}$. More specifically, for a Gaussian unitary that transforms the annihilation operators as

$$\hat{G}^\dagger\hat{\boldsymbol{a}}\hat{G} = X\hat{\boldsymbol{a}} + Y\hat{\boldsymbol{a}}^\dagger + \boldsymbol{\beta}, \tag{5}$$

where $\hat{\boldsymbol{a}} = (\hat{a}_1, \ldots, \hat{a}_M)$, we have

$$\boldsymbol{\alpha}' = X^\dagger\boldsymbol{\alpha} - Y^{\mathrm{T}}\boldsymbol{\alpha}^*. \tag{6}$$

Gaussian measurements are the ones whose POVMs are given as [32,81]

$$\hat{\Pi}(\boldsymbol{\alpha}) = \frac{1}{\pi^M}\hat{D}(\boldsymbol{\alpha})\hat{\rho}_G\hat{D}^\dagger(\boldsymbol{\alpha}), \tag{7}$$

where $\hat{\rho}_G$ is a Gaussian state that defines the Gaussian measurement. Gaussian measurements include homodyne, heterodyne, and general-dyne detections, which are frequently employed in current quantum optical circuits. Without loss of generality, the Gaussian state $\hat{\rho}_G$ that defines the measurement can be assumed to be a pure Gaussian state $|\psi_G\rangle\langle\psi_G|$ because the Gaussian measurements represented by a mixed Gaussian state can be simulated by those with a pure Gaussian state. Thus, we write the Gaussian measurements' POVMs as

$$\hat{\Pi}(\boldsymbol{\alpha}) = \frac{1}{\pi^M}\hat{D}(\boldsymbol{\alpha})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\alpha}). \tag{8}$$

The completeness of POVMs is guaranteed by the twirling property of the displacement operator, Eq. (2).

B. The median-of-means estimator

Finally, since all the estimators we use for our classical algorithms to solve the quantum mean-value problem are based on the median-of-means estimator [77,82], let us provide the relevant parameters. Suppose we want to estimate the mean of a random variable $X(\boldsymbol{\alpha})$,

$$\mathbb{E}[X] = \int d^{2M}\boldsymbol{\alpha}\, p(\boldsymbol{\alpha})X(\boldsymbol{\alpha}), \tag{9}$$

where $p(\boldsymbol{\alpha})$ is a proper probability distribution, i.e., $\mathbb{E}[X]$ is the average of the random variable $X(\boldsymbol{\alpha})$ over $\boldsymbol{\alpha} \sim p(\boldsymbol{\alpha})$. Let $\mathrm{Var}(\boldsymbol{\alpha})$ be the variance of the random variable $X(\boldsymbol{\alpha})$. Then, the median-of-means estimator is an estimator μ constructed by taking the median of means of subsamples, and, importantly, it satisfies

$$\Pr[|\mu - \mathbb{E}[X] \geq \epsilon] \leq 1 - \delta, \tag{10}$$

if we use N samples, where $N \geq 544\mathrm{Var}(X)\epsilon^{-2}\log(1/\delta)$ [77,82]. Hence, to bound the sample complexity, it suffices to find an upper bound on the variance (or the second moment) of the random variable X.

C. Sampling by chain rule

Finally, our classical algorithms are randomized algorithms based on sampling using the chain rule of conditional probabilities. More specifically, when we have a probability distribution $p(\boldsymbol{n})$, one way to sample $\boldsymbol{n}$ from the probability distribution is to use the chain rule of the conditional probability,

$$p(\boldsymbol{n}) = p(n_1)p(n_2|n_1)\cdots p(n_M|n_1, \ldots, n_{M-1}). \tag{11}$$

Hence, if one can compute the marginal probability, by sampling each random variable n_i using the marginal probabilities, one can sample $\boldsymbol{n}$ from the full probability distribution $p(\boldsymbol{n})$ [83]. Therefore, the complexity of computing the marginal probabilities determines that of sampling.

IV. PROBLEM SETUP

For computational tasks, we will examine two problems: (i) the sampling problem and (ii) the quantum mean-value problem. We now provide precise definitions of these two problems.

The sampling problem is defined as simulating the output probability distribution of a given quantum circuit, which is the basis of the current quantum-advantage experiments, such as random circuit sampling [3–6] and boson sampling [7–12]. Its definition is as follows:

Definition 1 (Sampling problem). Given as input an efficient description of an $(M + L)$-mode product input state $|\psi\rangle$, a quantum circuit $\mathcal{E}$ acting on $(M + L)$ modes that performs measurements on L modes with adaptive control, and a measurement on the remaining M output modes described by a POVM $\{\Pi_{\boldsymbol{x}}\}$, generate output samples according to the probability distribution $p(\boldsymbol{x}) \equiv \mathrm{Tr}[\Pi_{\boldsymbol{x}}\mathcal{E}(|\psi\rangle\langle\psi|)]$.

In the definition, two parameters, the number of output modes M and the number of modes that are used for measurement for adaptive control L, appear. When $L = 0$, there is no adaptive control in the circuit. Our main focus in this work is to investigate for which scalings of L as a function of M the corresponding sampling problem admits an efficient classical simulation, i.e., can be

solved in time polynomial in M, depending on the class of quantum circuits $\mathcal{E}$.

Note that in our definition, we choose the notion of sampling as exact sampling; hence, it is the hardest for classical computers (compared to approximate sampling).

For the quantum mean-value problem, the goal is to estimate the expectation value of a product operator $\hat{O} = \hat{O}_1 \otimes \cdots \otimes \hat{O}_M$ for an output state of a quantum circuit, $|\psi_{\text{out}}\rangle$, with an additive error,

$$\langle\psi_{\text{out}}|\hat{O}|\psi_{\text{out}}\rangle. \tag{12}$$

For a finer complexity analysis, we consider a product structure $\hat{O} = \hat{O}_A \otimes \hat{\mathbb{1}}_B$. Thus,

$$\langle\psi_{\text{out}}|\hat{O}_A \otimes \hat{\mathbb{1}}_B|\psi_{\text{out}}\rangle, \tag{13}$$

where $\hat{O}_A$ is the nontrivial part of the observable on subsystem A, still assumed to be in a product form, i.e., $\hat{O}_A = \prod_{i\in A} \hat{O}_i$. We let l denote the number of modes in the subsystem A.

Definition 2 (Quantum mean-value problem). Given as input an efficient description of an $(M + L)$-mode product input state $|\psi\rangle$, a quantum circuit $\mathcal{E}$ acting on $(M + L)$ modes that performs measurements on L modes with adaptive control, and an M-mode product observable $\hat{O} = \hat{O}_A \otimes \hat{\mathbb{1}}_B$, estimate the expectation value of $\hat{O}$ with respect to the output state of the circuit, $\text{Tr}[\hat{O}\mathcal{E}(|\psi\rangle\langle\psi|)]$ to within additive error ϵ with success probability at least $1 - \delta$.

In this setting, the relevant size parameters are again the number of output modes M and the number L of modes used for adaptive control. Our goal in this work is to identify regimes of L and classes of observables $\hat{O}_A$ for which this quantum mean-value problem admits an efficient classical algorithm, namely one whose running time is polynomial in M, $1/\epsilon$, and $\log(1/\delta)$.

For quantum circuits, we will focus on the following three different families: (i) Gaussian circuits without feedforward, (ii) Gaussian circuits augmented by photon-number measurement and adaptive Gaussian unitary gates depending on the photon-number measurement outcomes, (iii) Gaussian circuits augmented by Gaussian measurement and adaptive Gaussian unitary gates depending on the Gaussian measurement outcomes. Here, the choice of the input states and measurements depends on the tasks. The main reason that we investigate these three families is the feasibility of their implementations in practice. One may notice that we assume that all gate elements, including adaptive operations, are composed of Gaussian gates, which are experimentally accessible. Hence, for the hardness results, non-Gaussian elements, necessary resources for the hardness, are introduced through the input states and measurements.

Whenever the underlying circuit model is universal for quantum computation, in the sense that it can realize any polynomial-size adaptive circuit, the corresponding quantum mean-value problem is BQP-complete: additive-error estimation of suitable observables already captures the full power of a universal quantum computer. In our measurement-adaptive Gaussian architecture, this universality can be achieved by allowing the number L of adaptive measurements to grow polynomially with the system size [30,34], whereas our main results show that for any fixed constant L the quantum mean-value problem for a broad class of observables remains efficiently classically simulable.

For the sampling problem, we restrict to Gaussian input states and Gaussian measurements; otherwise, the corresponding sampling problems are already known to be hard to classically simulate due to boson sampling's hardness results [27–29,84]. Under this restriction, the first and third circuit families are already known to be efficiently simulable, since all elements are Gaussian [85]. In contrast, the second family is still nontrivial because it necessarily involves non-Gaussian elements. On the other hand, for the quantum mean-value problem, a general product input state and measurement, which may be non-Gaussian, is considered.

Finally, for simplicity, we will frequently assume that relevant quantum states (not for Gaussian states) have a finite maximum photon number,

$$|\psi\rangle = \bigotimes_{i=1}^{M} \left(\sum_{m_i=0}^{n_{\max}} a_{m_i}^{(i)} |m_i\rangle \right), \tag{14}$$

where $|m_i\rangle$ is the Fock state with photon number m_i and $a_{m_i}^{(i)}$ is the coefficient of the ith mode's m_i Fock state. When a given state does not satisfy this, we can approximate it by truncating the state by choosing the maximum photon number $n_{\max}$ appropriately. Although it induces an approximation error, such an approximation error is negligible if we choose $n_{\max}$ sufficiently large. Since the high photon occupation probability decays quickly in typical physical states, $n_{\max}$ does not have to be high in practice. For example, the system with dual-rail qubit encodings, such as the KLM protocol, requires only $n_{\max} = 1$, and few-photon Fock states or weakly squeezed states may not require more than $n_{\max}$ larger than 10 in general. As an example, Ref. [24], simulating large-scale Gaussian boson-sampling experiments, uses $n_{\max} = 7$–10 to suppress the truncation error.

V. CLASSICAL ALGORITHMS

A. Gaussian circuits without feedforward

In many optical quantum circuits, Gaussian circuits are relatively easy to implement, but not sufficient for universal quantum computation. Hence, additional resources are introduced through non-Gaussian state preparation, non-Gaussian measurements, and measurement and feedforward [30,37–39,41,42]. In this section, however, we investigate Gaussian circuits without any measurement and feedforward operations, but allow arbitrary input states and measurements, and we consider the associated computational tasks.

1. Sampling

For sampling, we consider the following configuration:

(1) Input state: product state;
(2) Quantum circuit: Gaussian unitary circuit;
(3) Measurement: Gaussian measurement.

As emphasized in the previous section, since such a configuration can realize boson-sampling circuits [27–29], classical simulation of their sampling behavior is known to be hard. On the other hand, if the input state is restricted to Gaussian states and hence the entire circuit is Gaussian, it is well established that such circuits are efficiently classically simulable [31–33].

2. Quantum mean-value problem

We now consider the quantum mean-value problem in this setup with the following configuration:

(1) Input state: product state;
(2) Quantum circuit: Gaussian unitary circuit;
(3) Observable: product observable.

Thus, we consider approximation of $\langle\psi|\hat{G}^\dagger\hat{O}\hat{G}|\psi\rangle$, where $\hat{G}$ is a Gaussian unitary circuit, $|\psi\rangle$ is an arbitrary product state, and $\hat{O}=\hat{O}_A\otimes\hat{\mathbb{1}}_B$ is a product observable: For linear-optical circuits, it was proven in Ref. [77] that estimating the expectation value within additive error ϵ with success probability $1-\delta$ requires $O(M^2\|\hat{O}_A\|_2^2\mathrm{Tr}[\hat{\rho}_A^2]\log(1/\delta)/\epsilon^2)$ computational cost, where $\hat{\rho}_A\equiv\mathrm{Tr}_B[\hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger]$ is the reduced density matrix on the system A. Here we extend this result from linear-optical circuits to general Gaussian circuits:

Theorem 1 (Quantum mean-value problem in Gaussian circuits without feedforward). Consider an M-mode Gaussian circuit $\hat{G}$ and an arbitrary product input state $|\psi\rangle$, and an M-mode product operator $\hat{O}=\hat{O}_A\otimes\hat{\mathbb{1}}_B$. The expectation value $\langle\psi|\hat{G}^\dagger\hat{O}\hat{G}|\psi\rangle$ can be approximated within additive error ϵ with probability $1-\delta$ in running time

$$O\left(\frac{M^2\|\hat{O}_A\|_2^2\mathrm{Tr}[\hat{\rho}_A^2]\log(1/\delta)}{\epsilon^2}\right), \tag{15}$$

where $\hat{\rho}_A\equiv\mathrm{Tr}_B[\hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger]$ is the reduced density matrix on the system A.

Proof sketch. Details are provided in Appendix A 1. Using Eq. (1), the mean value can be rewritten as

$$\langle\psi|\hat{G}^\dagger(\hat{O}_A\otimes\hat{\mathbb{1}}_B)\hat{G}|\psi\rangle=\int d^{2l}\boldsymbol{\alpha}_A\,p(\boldsymbol{\alpha}_A)X(\boldsymbol{\alpha}_A), \tag{16}$$

where l is the number of modes in the subsystem A, and we defined a probability distribution and a random variable,

$$p(\boldsymbol{\alpha}_A)=\frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l\|\hat{O}_A\|_2^2},\quad X(\boldsymbol{\alpha}_A)=\frac{\chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A)\,\|\hat{O}_A\|_2^2}{\chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)}, \tag{17}$$

respectively. By the twirling property, Eq. (2), $p(\boldsymbol{\alpha}_A)$ is a normalized probability distribution. Thus, the expectation value equals the mean of the random variable $X(\boldsymbol{\alpha}_A)$, which allows us to employ the median-of-means estimator.

By direct calculation, we show that the variance of X is bounded above by $\|\hat{O}_A\|_2^2\|\hat{\rho}_A\|_2^2$. Moreover, the product structure of $p(\boldsymbol{\alpha}_A)$ and Eq. (4) guarantee that sampling $\boldsymbol{\alpha}_A\sim p(\boldsymbol{\alpha}_A)$ and evaluating $X(\boldsymbol{\alpha}_A)$ can be done efficiently, respectively. Explicitly, they can be written in a product structure,

$$p(\boldsymbol{\alpha}_A)=\prod_{i\in A}\frac{\left|\chi_{\hat{O}_i}(\alpha_i)\right|^2}{\pi\|\hat{O}_i\|_2^2}, \tag{18}$$

$$X(\boldsymbol{\alpha}_A)=\prod_{i=1}^{M}\chi^*_{\psi_i}(\alpha_i')\prod_{i\in A}\frac{\|\hat{O}_i\|_2^2}{\chi^*_{\hat{O}_i}(\alpha_i)}, \tag{19}$$

where $\boldsymbol{\alpha}'$ is defined by

$$\hat{D}^\dagger(\boldsymbol{\alpha}')=\hat{G}^\dagger\big(\hat{D}^\dagger(\boldsymbol{\alpha}_A)\otimes\hat{D}^\dagger(\mathbf{0}_B)\big)\hat{G}, \tag{20}$$

and computed in $O(lM)$. Hence, applying the median-of-means estimator Eq. (10) with the above variance bound then yields the claimed complexity. ■

Hence, the algorithm runs as follows:

(1) Sample $\boldsymbol{\alpha}_A\in\mathbb{C}^l$ from $p(\boldsymbol{\alpha}_A)$ by independently sampling α_i from $|\chi_{\hat{O}_i}(\alpha_i)|^2/(\pi\|\hat{O}_i\|_2^2)$;
(2) Compute $X(\boldsymbol{\alpha}_A)$ for the sampled $\boldsymbol{\alpha}_A$ by using Eq. (6);

(3) Apply the median-of-mean estimator to estimate the expectation value.

Here, for sampling from $|\chi_{\hat{O}_i}(\alpha_i)|^2/(\pi\|\hat{O}_i\|_2^2)$, one may consider various well-known classical algorithms. For example, for general purposes rejection sampling and Markov Chain Monte Carlo algorithms are natural choices to perform the sampling [86] since we can efficiently compute the probability density.

On the other hand, Theorem 1 shows that whenever $\|\hat{O}_A\|_2^2 = O(\text{poly}(M))$, there exists a classical algorithm that efficiently solves the quantum mean-value problem for Gaussian circuits. This condition is naturally met in many relevant scenarios, such as local observables with support on $l = O(1)$ modes and low-rank (possibly global) observables whose Hilbert-Schmidt norm remains polynomially bounded. The former is particularly relevant for near-term applications that focus on estimating local quantities (e.g., VQAs [55–60]), as discussed in the Introduction.

We note that the dependence of the classical complexity on the number of modes l in the subsystem A enters mainly through the Hilbert-Schmidt norm $\|\hat{O}_A\|_2^2$ and the local purity $\text{Tr}[\hat{\rho}_A^2]$. When $l = \omega(1)$, the Hilbert-Schmidt norm can become extensive or even grow exponentially with l (for instance, for highly nonlocal observables with large support), which in turn can make the runtime exponentially large. In contrast, for genuinely local few-body observables or low-rank observables with bounded spectrum, $\|\hat{O}_A\|_2^2$ typically remains polynomial in M, so that this potential source of hardness is absent. On the other hand, if the output state is highly entangled, the purity of the reduced state $\text{Tr}[\hat{\rho}_A^2]$ can be significantly suppressed. This suppression always reduces the complexity, but for typical highly entangled states, one still expects at most an inverse-polynomial suppression in M since typical highly entangled states' purity is at least inverse-polynomial in the system size M [87,88].

We emphasize that the input state may be arbitrary as long as it is a product state. Hence, even if the input state has a large amount of non-Gaussian resources, such as a high stellar rank [89], the corresponding quantum mean-value problem is still easy, which is in stark contrast to the sampling problem.

B. Photon-number-measurement-adaptive Gaussian circuits

The Gaussian circuits considered in the preceding section are not believed to be as powerful as a universal quantum computer, even when supplemented with non-Gaussian inputs or measurements [27]. Consequently, in many photonic architectures, universal quantum computation is pursued through intermediate measurements and feedforward, as the direct implementation of non-Gaussian gates is often experimentally demanding. For feasibility, we assume that the feedforward unitaries are Gaussian.

In this section, we consider photon-number-resolving detections for feedforward operations. In other words, conditioned on the detected photon numbers, a measurement-outcome-dependent Gaussian operation is applied. We now analyze the computational complexity of simulating such circuits for both sampling and mean-value estimation, with particular emphasis on the dependence on the number of measurements and adaptive steps.

1. Measurement and feedforward

We now formulate circuits augmented with measurement and feedforward (see Fig. 1). The circuit consists of a computing register and a measurement register: the modes in the computing register are retained until the final measurement, while the modes in the measurement register are measured in the middle, and the outcomes are used to determine feedforward operations on either the computing or measurement registers.

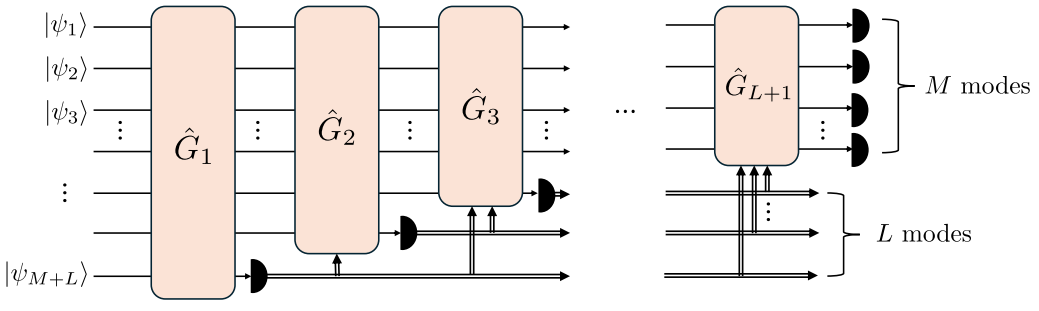


FIG. 1. Schematics of quantum circuits of interest in this work. The input state is a product state, and the quantum circuits are composed of Gaussian circuits, measurements, and feedforward operations. More specifically, we consider photon-number measurements (Sec. V B) and Gaussian measurements (Sec. V C) for feedforward operations. The final measurement on the computing register is always Gaussian for sampling and implicit for the quantum mean-value problem, depending on the observable. Note that the final measurement can also be adaptive through the final circuit layer $\hat{G}_{L+1}$.

We begin with a product input state $|\psi\rangle$ and apply a Gaussian unitary $\hat{G}^{(1)}$,

$$|\psi^{(1)}\rangle = \hat{G}^{(1)}|\psi\rangle. \tag{21}$$

Next, one of the modes is measured using a rank-one POVM $\{|n_1\rangle\langle n_1|\}_{n_1}$, and an adaptive Gaussian unitary is applied to the remaining modes,

$$|\psi^{(1)}\rangle \to \hat{G}^{(2)}_{n_1}(\hat{\mathbb{1}} \otimes \langle n_1|)|\psi^{(1)}\rangle \equiv |\tilde{\psi}^{(2)}_{n_1}\rangle, \tag{22}$$

where $|\tilde{\psi}^{(2)}_{n_1}\rangle$ is a unnormalized state. More generally, at the kth step, we have

$$|\tilde{\psi}^{(k)}_{n_1,\ldots,n_{k-1}}\rangle \to \hat{G}^{(k+1)}_{n_1,\ldots,n_k}(\hat{\mathbb{1}} \otimes \langle n_k|)|\tilde{\psi}^{(k)}_{n_1,\ldots,n_{k-1}}\rangle \tag{23}$$

$$\equiv |\tilde{\psi}^{(k+1)}_{n_1,\ldots,n_k}\rangle. \tag{24}$$

The intermediate (unnormalized) state can be rewritten as

$$|\tilde{\psi}^{(k+1)}_{n_1,\ldots,n_k}\rangle = \langle n_1,\ldots,n_k|\hat{G}^{(k+1)}_{n_1,\ldots,n_k}\cdots\hat{G}^{(2)}_{n_1}\hat{G}^{(1)}|\psi\rangle \tag{25}$$

$$\equiv \langle n_1,\ldots,n_k|\hat{G}_{n_1,\ldots,n_k}|\psi\rangle \tag{26}$$

$$\equiv \langle n_1,\ldots,n_k|\psi_{n_1,\ldots,n_k}\rangle, \tag{27}$$

and the probability of observing outcome $(n_1,\ldots,n_k)$ is

$$p(n_1,\ldots,n_k) \equiv \langle\tilde{\psi}^{(k+1)}_{n_1,\ldots,n_k}|\tilde{\psi}^{(k+1)}_{n_1,\ldots,n_k}\rangle. \tag{28}$$

After L steps of measurement and adaptive operations, the final state is written as

$$|\tilde{\psi}^{(L+1)}_{n_1,\ldots,n_L}\rangle, \tag{29}$$

and the corresponding probability distribution factorizes as

$$p(n_1,\ldots,n_L) = p(n_1)p(n_2|n_1)\cdots p(n_L|n_1,\ldots,n_{L-1}). \tag{30}$$

By measuring the final quantum state $|\tilde{\psi}^{(L+1)}_{n_1,\ldots,n_L}\rangle$, the computation is completed. Now, let us consider the sampling problem and the quantum mean-value problem, respectively.

2. Sampling

We now consider a Gaussian input state and a Gaussian measurement at the final stage. Thus, the configuration is as follows:

(1) Input state: Gaussian input state;
(2) Quantum circuit: Gaussian unitary circuit with L adaptive photon-number measurements;
(3) Final measurement: Gaussian measurement.

This restriction is natural because if we allow a non-Gaussian measurement, then even without mid-circuit measurements, the classical simulation of sampling is already hard due to the hardness of boson sampling [27–29]. Similarly, if we allow a non-Gaussian input with Gaussian measurements, the resulting configuration has also been proven to be classically hard [84]. Therefore, throughout this subsection, we assume Gaussian inputs and Gaussian final measurements.

Let L denote the number of modes used for adaptive photon-number measurements. We prove that when $L = O(\log M)$, the sampling problem can be simulated efficiently. More specifically, since the intermediate measurements that are not used for Gaussian adaptive feedforward can be deferred to the end of the circuit, we prove that the sampling problem can be efficiently simulated if the output measurement is composed of M Gaussian measurements and L photon-number resolving measurements, where the latter can be used for Gaussian feedforward.

Theorem 2 (Sampling in Gaussian circuits with photon-number measurement and Gaussian feedforward). Consider an $(M+L)$-mode Gaussian input state, an $(M+L)$-mode Gaussian circuit, and a measurement scheme on the output modes that consists of (i) photon-number-resolving measurements on at most L modes and (ii) Gaussian measurements on the remaining M modes. The photon-number measurements may be performed adaptively, in the sense that arbitrary Gaussian operations may depend on the previous outcomes (Gaussian feedforward). If $L = O(\log M)$, then the sampling from the resulting output distribution can be efficiently simulated on a classical computer.

Proof. The output probability distribution of the final Gaussian measurement $\{\hat{\Pi}(\boldsymbol{\alpha})\}$ under adaptive operations on M modes can be written as

$$p(\boldsymbol{\alpha}) = \sum_{\boldsymbol{n}} p(\boldsymbol{n})\, p(\boldsymbol{\alpha}|\boldsymbol{n}) \tag{31}$$

$$= \sum_{\boldsymbol{n}} p(\boldsymbol{n}) \frac{\langle\psi_{\boldsymbol{n}}|(\hat{\Pi}(\boldsymbol{\alpha}) \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle}{\langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}}_M \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle}, \tag{32}$$

where $\boldsymbol{n} \equiv (n_1,\ldots,n_L)$, $|\psi_{\boldsymbol{n}}\rangle \equiv \hat{G}_{\boldsymbol{n}}|\psi\rangle$, and

$$p(\boldsymbol{n}) = \langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}}_M \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle. \tag{33}$$

Here, $\hat{G}_{\boldsymbol{n}}$ denotes the feedforward Gaussian circuit conditioned on outcome $\boldsymbol{n}$, and $\hat{\Pi}(\boldsymbol{\alpha})$ is a Gaussian POVM.

Hence, $p(\boldsymbol{n})$ is normalized to 1 as

$$1 = \sum_{\boldsymbol{n}} p(\boldsymbol{n}) = \sum_{\boldsymbol{n}} \langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}}_M \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle = \sum_{\boldsymbol{n}} \langle\tilde{\psi}_{\boldsymbol{n}}|\tilde{\psi}_{\boldsymbol{n}}\rangle, \tag{34}$$

where $|\tilde{\psi}_{\boldsymbol{n}}\rangle \equiv \langle\boldsymbol{n}|\psi_{\boldsymbol{n}}\rangle$ is the unnormalized postmeasurement state obtained when the measurement for feedforward is $\boldsymbol{n}$.

Our classical sampling algorithm proceeds as follows. First, sample $\boldsymbol{n}$ from $p(\boldsymbol{n})$ using a chain-rule-based algorithm,

$$p(\boldsymbol{n}) = p(n_1)p(n_2|n_1)\cdots p(n_L|n_1,\ldots,n_{L-1}), \tag{35}$$

where the computational cost mainly depends on the complexity of computing the marginal probabilities $p(n_1,\ldots,n_k)$. Next, for the obtained $\boldsymbol{n}$, sample $\boldsymbol{\alpha}$ from

$$p(\boldsymbol{\alpha}|\boldsymbol{n}) = \frac{\langle\psi_{\boldsymbol{n}}|(\hat{\Pi}(\boldsymbol{\alpha}) \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle}{\langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}}_M \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle}, \tag{36}$$

which completes the procedure.

The computational bottleneck is thus (i) computing the marginal probabilities of $p(\boldsymbol{n})$ and (ii) evaluating $\langle\psi_{\boldsymbol{n}}|(\hat{\Pi}(\boldsymbol{\alpha}) \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle$. Since $|\psi_{\boldsymbol{n}}\rangle$ is Gaussian, these quantities reduce to evaluating loop hafnians [19,28], whose cost scales exponentially in the number of measured modes L [20]. Because $\hat{\Pi}(\boldsymbol{\alpha})$ is Gaussian, the numerator also reduces to a loop hafnian of size L.

Therefore, the overall simulation is efficient whenever $L = O(\log M)$, establishing the theorem. ■

Hence, in summary, the algorithm runs as follows:

(1) Sample $\boldsymbol{n}$ from $p(\boldsymbol{n})$ by using the chain-rule-based algorithm, where $p(\boldsymbol{n})$'s marginal probabilities are given by loop hafnians [19,28];
(2) Sample $\boldsymbol{\alpha}$ from $p(\boldsymbol{\alpha}|\boldsymbol{n})$ for a given $\boldsymbol{n}$ by using the chain-rule-based algorithm, which again reduces to computing loop hafnians.

3. Quantum mean-value problem

We now turn to the quantum mean-value problem in this circuit configuration. Thus, we now consider the quantum mean-value problem in this setup with the following configuration:

(1) Input state: product state;
(2) Quantum circuit: Gaussian unitary circuit with L adaptive photon-number measurements;
(3) Observable: product observable.

Here, we assume a general product state as the input, and the measurement is implicit since the goal is to estimate the expectation value of an observable $\hat{O}$. We show that if the number of adaptive measurements L is constant, i.e., $L = O(1)$, the observable can be efficiently estimated on a classical computer, provided the Hilbert-Schmidt norm of the operator $\hat{O}$ is at most polynomially large:

Theorem 3 (Quantum mean-value problem in Gaussian circuits with photon-number measurement and Gaussian feedforward). Consider an arbitrary product input state and a Gaussian circuit, augmented by photon-number measurements and Gaussian feedforward, which outputs an M-mode quantum state, and assume that the input state has a fixed maximum photon number. If the number of photon-number measurements used for feedforward is $L = O(1)$, then the expectation value of an M-mode observable $\hat{O} = \hat{O}_A \otimes \hat{\mathbb{1}}_B$ can be approximated within additive error ϵ and success probability $1-\delta$ in running time,

$$O\left(\frac{\mathrm{poly}(M)\|\hat{O}_A\|_2^2 \mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{n}}^2)]\log(1/\delta)}{\epsilon^2}\right), \tag{37}$$

where $\hat{\rho}_{\boldsymbol{n}}$ is the reduced density matrix on the system A when the photon-number measurement outcome is $\boldsymbol{n}$ and $\mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{n}}^2)]$ is the average purity.

Again, we emphasize that because the input can be an arbitrary product state, it may have a large non-Gaussian resource. However, this resource does not change the complexity, unlike sampling; instead, the number of measurements for feedforward drastically changes the complexity.

Finally, we remark that, as in the sampling task, if the input state is Gaussian instead of an arbitrary state, then sampling from $p(\boldsymbol{n})$ and computing $X(\boldsymbol{n},\boldsymbol{\alpha}_A)$ that appear in the proof remain efficient up to $L = O(\log M)$ because computing $p(\boldsymbol{n})$ and $X(\boldsymbol{n},\boldsymbol{\alpha}_A)$ reduces to computing (loop-)hafnians associated to L-mode Gaussian states, which can be computed in $O(\exp(L))$ [20].

Proof sketch. Details are provided in Appendix A 2. Consider a general product input $|\psi\rangle$. The probability of obtaining measurement outcome $\boldsymbol{n}$ is then written as

$$p(\boldsymbol{n}) = \langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}} \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle, \tag{38}$$

where $|\psi_{\boldsymbol{n}}\rangle \equiv \hat{G}_{\boldsymbol{n}}|\psi\rangle$ is the (unnormalized) postmeasurement state for outcome $\boldsymbol{n}$.

For a given measurement outcome $\boldsymbol{n}$, the conditional expectation value of $\hat{O} = \hat{O}_A \otimes \hat{\mathbb{1}}_B$ is given by

$$\frac{\langle\psi|\hat{G}_{\boldsymbol{n}}^\dagger(\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)\hat{G}_{\boldsymbol{n}}|\psi\rangle}{\langle\psi|\hat{G}_{\boldsymbol{n}}^\dagger(\hat{\mathbb{1}}_A \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle\boldsymbol{n}|)\hat{G}_{\boldsymbol{n}}|\psi\rangle}, \tag{39}$$

and averaging over all outcomes gives

$$\langle \hat{O} \rangle = \sum_{\boldsymbol{n}} \langle \psi | \hat{G}_{\boldsymbol{n}}^{\dagger} (\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) \hat{G}_{\boldsymbol{n}} | \psi \rangle, \tag{40}$$

which is thus the expectation value we want to estimate. We now rewrite the expectation value of $\hat{O}$ as

$$\langle \hat{O} \rangle = \int d^{2l}\boldsymbol{\alpha}_A \sum_{\boldsymbol{n}} p(\boldsymbol{n})\, q(\boldsymbol{\alpha}_A)\, X(\boldsymbol{n}, \boldsymbol{\alpha}_A), \tag{41}$$

where l is the number of modes in the subsystem A, and we defined a probability distribution and a random variable,

$$q(\boldsymbol{\alpha}_A) \equiv \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l \|\hat{O}_A\|_2^2}, \tag{42}$$

$$X(\boldsymbol{n}, \boldsymbol{\alpha}_A) \equiv \frac{\|\hat{O}_A\|_2^2 \langle \psi_{\boldsymbol{n}} | (\hat{D}^{\dagger}(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle}{p(\boldsymbol{n}) \chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)}, \tag{43}$$

respectively. Here, the normalization of $q(\boldsymbol{\alpha}_A)$ follows from the twirling property of displacement operators, Eq. (2). Due to the above expression of the expectation value, our algorithm estimates $\langle \hat{O} \rangle$ by sampling $\boldsymbol{n} \sim p(\boldsymbol{n})$ and $\boldsymbol{\alpha}_A \sim q(\boldsymbol{\alpha}_A)$, and then applying the median-of-means estimator to $X(\boldsymbol{n}, \boldsymbol{\alpha}_A)$.

By direct calculation, we upper-bound the variance of X by $\|\hat{O}_A\|_2^2 \mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{n}}^2)]$, where $\hat{\rho}_{\boldsymbol{n}}$ is the reduced density matrix on subsystem A for outcome $\boldsymbol{n}$.

Finally, efficient implementation requires analyzing three subroutines: (i) sampling from $p(\boldsymbol{n})$, (ii) sampling from $q(\boldsymbol{\alpha}_A)$, and (iii) computing $X(\boldsymbol{n}, \boldsymbol{\alpha}_A)$.

Sampling from $q(\boldsymbol{\alpha}_A)$ is straightforward due to its product structure, cf. Eq. (18). Sampling from $p(\boldsymbol{n})$ amounts to computing marginal photon-number probabilities of a state obtained by applying a Gaussian circuit to a general product input. This is nontrivial in general. For Fock input states and linear-optical circuits, Gurvits introduced an algorithm (the so-called Gurvits' second algorithm) that is efficient whenever the number of marginal modes is constant [27,61]. We extend this algorithm to arbitrary product input states and Gaussian circuits (see Sec. VI and Appendix F for more details):

Lemma 1 (Computing photon-number marginal probabilities in Gaussian circuits). Consider an M-mode bosonic system, product states $|\psi\rangle$, $|\phi\rangle$, a Gaussian unitary $\hat{G}_0$, and an L-mode Fock state $|\boldsymbol{n}\rangle$. Here, we assume that the states $|\phi\rangle$ and $|\psi\rangle$ have a fixed maximum photon number. Then

$$\langle \phi | \hat{G}_0^{\dagger} (\hat{\mathbb{1}} \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) \hat{G}_0 | \psi \rangle \tag{44}$$

can be exactly computed in polynomial time in M if $L = O(1)$.

Note that $|\psi\rangle$ and $|\phi\rangle$ may differ, which is an additional generalization beyond Gurvits' original setting. Using this lemma, we can efficiently sample from $p(\boldsymbol{n})$ whenever $L = O(1)$.

Finally, evaluating $X(\boldsymbol{n}, \boldsymbol{\alpha}_A)$ requires computing a similar overlap term. Since the lemma also applies when $|\psi\rangle$ and $|\phi\rangle$ differ, we can absorb $\hat{D}^{\dagger}(\boldsymbol{\alpha}_A)$ into the input and compute the random variable $X(\boldsymbol{n}, \boldsymbol{\alpha}_A)$ efficiently when $L = O(1)$.

This completes the proof of the theorem. More detailed analysis can be found in Sec. VI and Appendix F. ■

Thus, in summary, the algorithm runs as follows:

(1) Sample $\boldsymbol{\alpha}_A \in \mathbb{C}^l$ from $|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)|^2$ by sampling α_i independently from $|\chi_{\hat{O}_i}(\alpha_i)|^2$;
(2) Given $\boldsymbol{\alpha}_A$, we compute $X(\boldsymbol{n}, \boldsymbol{\alpha}_A)$ by invoking Lemma 1;
(3) Apply the median-of-means estimator for the collected samples.

As in Theorem 1, the dependence on the system size A, l, is implicitly determined by the Hilbert-Schmidt norm of the operator $\hat{O}_A$ and the average purity. For example, the $l = O(1)$ case can be efficiently solved using our algorithm if $L = O(1)$, whereas $l = \omega(1)$ can be hard for our algorithm. We remark that this clearly suggests a transition between a setup with limited adaptivity and the universal quantum computer through increasing the adaptive steps because while the universal quantum computer, i.e., $L = \mathrm{poly}(M)$ with $l = O(1)$ is sufficient to solve BQP-complete problems, our theorem shows that the $l = O(1)$ case is easy for classical computer.

C. Gaussian-measurement-adaptive Gaussian circuits

In this section, instead of photon-number-measurement-based feedforward, we consider feedforward based on Gaussian measurements, which is more experimentally accessible. Although the operations remain Gaussian, it is known that if non-Gaussian input states are provided, Gaussian circuits with Gaussian-measurement-based feedforward and Gaussian final measurements are sufficient for universal quantum computation [34].

1. Sampling

Since this family includes, as a subset, the Gaussian circuits considered in the previous section, simulating the sampling problem with either general non-Gaussian input states or general non-Gaussian measurements is classically hard. On the other hand, if both the input states and final measurements are Gaussian, then the entire process remains Gaussian, and simulating such circuits is efficiently possible on a classical computer.

2. Quantum mean-value problem

Let us consider the quantum mean-value problem in this circuit configuration with a general product input $|\psi\rangle$. Thus, we now consider the quantum mean-value problem in this setup with the following configuration:

(1) Input state: product state;
(2) Quantum circuit: Gaussian unitary circuit with L adaptive Gaussian measurements;
(3) Observable: product observable.

We now show that the computational complexity for solving this problem is at most polynomial in the system size if the number of measurements for feedforward operations is $L = O(1)$:

Theorem 4 (Quantum mean-value problem in Gaussian circuit augmented by Gaussian measurement and Gaussian feedforward). Consider an arbitrary product input and a Gaussian circuit, augmented by Gaussian measurements and Gaussian feedforward, which outputs an M-mode quantum state, and assume that the input state has a fixed maximum photon number. If the number of Gaussian measurements for feedforward is at most constant, i.e., $L = O(1)$, the expectation value of an M-mode product observable $\hat{O} \equiv \hat{O}_A \otimes \hat{\mathbb{1}}_B$ can be approximated within additive error ϵ with probability $1 - \delta$ in running time,

$$O\left(\frac{\mathrm{poly}(M)\|\hat{O}_A\|_2^2 \mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{\beta}}^2)]\log(1/\delta)}{\epsilon^2}\right), \tag{45}$$

where $\hat{\rho}_{\boldsymbol{\beta}}$ is the reduced density matrix on the system A when the Gaussian measurement outcome is $\boldsymbol{\beta}$ and $\mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{\beta}}^2)]$ is the average purity.

Interestingly, even if the measurements for feedforward are Gaussian, the complexity increases as in the photon-number measurement case. While surprising, on a high level, this is consistent with a known result in Ref. [34], which demonstrates that Gaussian operations, including measurements and feedforward, are sufficient when GKP states are provided as input.

Proof sketch. The details of the proof are provided in Appendix A 3. First of all, the probability of obtaining $\boldsymbol{\beta}$ for feedforward is

$$p(\boldsymbol{\beta}) = \langle\psi_{\boldsymbol{\beta}}|(\hat{\mathbb{1}}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle, \tag{46}$$

where $|\psi_{\boldsymbol{\beta}}\rangle \equiv \hat{G}_{\boldsymbol{\beta}}|\psi\rangle$ is the (unnormalized) postmeasurement state upon measurement outcome $\boldsymbol{\beta}$ and $\{\hat{\Pi}_{\boldsymbol{\beta}} = \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta})/\pi^L\}$ is Gaussian POVMs. Thus, the expectation value when the outcome is $\boldsymbol{\beta}$ is

$$\frac{\langle\psi|\hat{G}_{\boldsymbol{\beta}}^\dagger(\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))\hat{G}_{\boldsymbol{\beta}}|\psi\rangle}{\langle\psi|\hat{G}_{\boldsymbol{\beta}}^\dagger(\hat{\mathbb{1}}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))\hat{G}_{\boldsymbol{\beta}}|\psi\rangle}. \tag{47}$$

If we take average over $\boldsymbol{\beta}$, we obtain the expectation value, which is written as

$$\langle\hat{O}\rangle = \int d^{2L}\boldsymbol{\beta}\langle\psi_{\boldsymbol{\beta}}|(\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle \tag{48}$$

$$= \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta}\, p(\boldsymbol{\beta})q(\boldsymbol{\alpha}_A)X(\boldsymbol{\beta},\boldsymbol{\alpha}_A), \tag{49}$$

where l is the number of modes in the subsystem A, and we defined a probability distribution

$$q(\boldsymbol{\alpha}_A) \equiv \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l\|\hat{O}_A\|_2^2}, \tag{50}$$

and the random variable

$$X(\boldsymbol{\beta},\boldsymbol{\alpha}_A) = \frac{\|\hat{O}_A\|_2^2\langle\psi_{\boldsymbol{\beta}}|(\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle}{p(\boldsymbol{\beta})\chi_A^*(\boldsymbol{\alpha}_A)}. \tag{51}$$

By direct calculation, we prove that the variance of the random variable is upper-bounded by $O(\|\hat{O}_A\|_2^2\mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{\beta}}^2)])$, where $\hat{\rho}_{\boldsymbol{\beta}}$ is the reduced density matrix on the Hilbert space A when the measurement outcomes are $\boldsymbol{\beta}$ and the average is taken over the probability distribution $p(\boldsymbol{\beta})$.

For the algorithm, we need to be able to (i) sample from $q(\boldsymbol{\alpha}_A)$, (ii) sample from $p(\boldsymbol{\beta})$, and (iii) compute $X(\boldsymbol{\beta},\boldsymbol{\alpha}_A)$. As before, (i) is easy because of the product structure of $\hat{O}_A$. Computing the marginal probability of $p(\boldsymbol{\beta})$ can be done efficiently when the number of modes involved in $\hat{\Pi}(\boldsymbol{\beta})$ is at most constant, i.e., $L = O(1)$ due to Lemma 1. Similarly, using Lemma 1, $X(\boldsymbol{\beta},\boldsymbol{\alpha}_A)$ can be efficiently computed when $L = O(1)$. Hence, the theorem. ■

Thus, in summary, the algorithm runs as follows:

(1) Sample $\boldsymbol{\alpha}_A \in \mathbb{C}^l$ from $|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)|^2$ by sampling α_i independently from $|\chi_{\hat{O}_i}(\alpha_i)|^2$;
(2) Given $\boldsymbol{\alpha}_A$, we compute $X(\boldsymbol{\beta},\boldsymbol{\alpha}_A)$ by invoking Lemma 1;
(3) Apply the median-of-means estimator for the collected samples.

As we have seen from the two previous theorems, Lemma 1 plays a crucial role in our classical algorithms. For the rest of this paper, we introduce the classical algorithm for the lemma.

VI. CLASSICAL ALGORITHMS USING LOW-RANK STRUCTURES

In this section, we present the details of Lemma 1, which plays a crucial role in our classical algorithms. First, we reduce the task of computing marginal probabilities in the photon-number basis to evaluating an associated generating function (see below for the precise definition). We then compute this generating function. As a first step, we show that the amplitudes of few-mode quantum states can be computed efficiently. Building on this, we demonstrate that the generating function corresponding to photon-number marginals over a small number of modes can likewise be computed efficiently on a classical computer. The latter can be viewed as an extension of Gurvits' second algorithm [27,61]. Conceptually, these results leverage the correspondence between few-mode structure and low-rank matrices, making the relevant quantities amenable to efficient classical computation or simulation. We expect that the results—and their proofs—are of independent interest in the study of computational complexity for quantum optical systems.

A. Computing the marginal probabilities of Gaussian circuits on photon-number basis

In this section, we show a way to compute the marginal probability in the photon-number basis for the output state generated by a Gaussian circuit $\hat{G}_0$ applied to a product state $|\psi\rangle$. Let us consider a general expression

$$q(\boldsymbol{n}) = \langle\phi|\hat{G}_0^\dagger|\boldsymbol{n}\rangle\langle\boldsymbol{n}|\hat{G}_0|\psi\rangle, \tag{52}$$

and its marginal

$$q(n_1,\ldots,n_L) = \sum_{n_{L+1},\ldots,n_M=0}^{\infty} q(\boldsymbol{n}) \tag{53}$$

$$= \langle\phi|\hat{G}_0^\dagger(|n_1,\ldots,n_L\rangle\langle n_1,\ldots,n_L| \otimes \hat{\mathbb{1}})\hat{G}_0|\psi\rangle. \tag{54}$$

One may see that when $|\phi\rangle = |\psi\rangle$, $q(\boldsymbol{n})$ represents the (marginal) probability in the photon-number basis. Thus, we call this function a marginal probability in a broader sense even when $|\psi\rangle \neq |\phi\rangle$. We have seen that this function plays a central role in the classical algorithms introduced in Sec. V. We present a general recipe to compute this function below. Note that we now assume that $|\psi\rangle$ and $|\phi\rangle$ have a maximum photon-number cutoff $n_{\max}$ for simplicity, as discussed in Sec. IV.

Before considering general cases, we first note that in the case of a Gaussian state $|\psi\rangle$, this is expressed simply as a loop hafnian with matrix size given by L [20,63,90] and thus the complexity is exponential in the number of modes. Thus, when L is at most logarithmic, we can compute this quantity in polynomial time. However, in more general cases, i.e., for non-Gaussian states, it may not be written as a simple loop hafnian. Remarkably, for Fock-state input $|\psi\rangle = |\boldsymbol{m}\rangle$ and linear-optical circuit, there is a well-known algorithm, so-called Gurvits' second algorithm, which enables us to compute $q(\boldsymbol{n})$ with complexity $O(N^{2L+1})$, where N is the total photon number of $|\boldsymbol{m}\rangle$ (see Sec. VI B 1) [27,61]. Hence, when $L = O(1)$, i.e., the number of marginal modes is at most constant, the corresponding quantity can be efficiently computed using classical computers. Our main contribution is to generalize this result for Fock states and linear-optical circuits to general product input states $|\phi\rangle$ and $|\psi\rangle$ and Gaussian circuits.

The main idea is to exploit the following Fourier relation [27,64] (see Appendix B for the details):

$$\tilde{G}(k) \equiv \langle\phi|\hat{G}_0^\dagger e^{-ik\theta\hat{\boldsymbol{n}}\cdot\boldsymbol{\omega}}\hat{G}_0|\psi\rangle, \tag{55}$$

$$G(\Omega) \equiv \sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty} q(\boldsymbol{n})\delta(\Omega - \boldsymbol{\omega}\cdot\boldsymbol{n}) = \frac{1}{\Omega_{\max}+1}\sum_{k=0}^{\Omega_{\max}} \tilde{G}(k)e^{ik\Omega}, \tag{56}$$

where $\hat{\boldsymbol{n}} \equiv (\hat{n}_1,\ldots,\hat{n}_M)$ is the photon number operator vector for M modes, where $\hat{n}_i \equiv \hat{a}_i^\dagger\hat{a}_i$, $k,\Omega \in \{0,\ldots,\Omega_{\max}\}$, $\theta \equiv 2\pi/(\Omega_{\max}+1)$, and $\boldsymbol{\omega} \in \mathbb{R}^M$. Thus, if we want to compute the marginal probability of $q(\boldsymbol{n})$ on the Fock basis up to the Lth mode, we set

$$\boldsymbol{\omega} = (\omega_1,\ldots,\omega_L,0,\ldots,0), \tag{57}$$

where $\omega_i = (n_{\max}+1)^{i-1}$. Then, each $\Omega \in \{0,\ldots,(n_{\max}+1)^L - 1\}$ corresponds to distinct marginal probabilities of $q(\boldsymbol{n})$ where $n_i \leq n_{\max}$. More explicitly, if we want to compute $q(n_1,\ldots,n_L)$, we set $\Omega = \sum_{i=1}^L \omega_i n_i$ with $\boldsymbol{\omega}$ defined above and compute the corresponding $G(\Omega)$, which is equivalent to $q(n_1,\ldots,n_L)$. Therefore, if we can compute $\tilde{G}(k)$, in principle, we can compute $G(\Omega)$ and thus its marginal probabilities. Here, because we need to sum k from 0 to $\Omega_{\max}$, the complexity increases as $O((n_{\max}+1)^L)$. With this in mind, computing $q(\boldsymbol{n})$ or its marginal probabilities boils down to exactly computing

$$\langle\phi|\hat{G}_0^\dagger\hat{P}(\boldsymbol{\varphi})\hat{G}_0|\psi\rangle \equiv \langle\phi|\hat{G}|\psi\rangle, \tag{58}$$

where $\hat{P}(\boldsymbol{\varphi}) \equiv e^{i\boldsymbol{\varphi}\cdot\hat{\boldsymbol{n}}}$ is a phase shifter with $(\varphi_1,\ldots,\varphi_L, 0,\ldots,0)$, which has exactly the same form of $\tilde{G}(k)$. We refer to this function as a generating function. Hence, the main complexity of this algorithm is attributed to computing $\tilde{G}(k)$.

B. Generalized Gurvits' second algorithm

In this section, we prove that

$$\tilde{G}(k) = \langle\phi|\hat{G}_0^\dagger\hat{P}(\boldsymbol{\varphi})\hat{G}_0|\psi\rangle \tag{59}$$

is easy to compute when $|\psi\rangle$ and $|\phi\rangle$ are product states and $\boldsymbol{\varphi}$ has at most constant nonzero phases. Without loss

of generality, let $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_L, 0, \ldots, 0)$ throughout this section.

As mentioned above, it has been known for Fock states and linear-optical circuits, $|\psi\rangle = |\phi\rangle = |\boldsymbol{n}\rangle$ and $\hat{G}_0 = \hat{U}$ is a linear-optical circuit, that $\tilde{G}(k)$ can be computed in a polynomial time when L is at most a constant:

Lemma 2 (Gurvits' second algorithm [27,61]). Consider an M-mode linear-optical circuit $\hat{U}$, a phase-shifter operator $\hat{P}(\boldsymbol{\varphi})$ with $\boldsymbol{\varphi} \equiv (\varphi_1, \ldots, \varphi_L, 0, \ldots, 0)$, and a Fock state $|\boldsymbol{n}\rangle$. If $L = O(1)$,

$$\langle \boldsymbol{n}|\hat{U}^\dagger \hat{P}(\boldsymbol{\varphi})\hat{U}|\boldsymbol{n}\rangle \tag{60}$$

can be exactly computed in polynomial time in the total photon number in $|\boldsymbol{n}\rangle$ and the number of modes M.

We generalize this to arbitrary product states $|\psi\rangle$ and $|\phi\rangle$, which may be different, and an arbitrary Gaussian circuit $\hat{G}_0$:

Lemma 3 (Generating function). Consider an M-mode bosonic system. Let $\hat{G}_0$ be a Gaussian unitary circuit and $\hat{P}(\boldsymbol{\varphi})$ be the phase shifter with phase $\boldsymbol{\varphi}$ and $|\psi\rangle$ and $|\phi\rangle$ be product states. If the number of nonzero elements in $\boldsymbol{\varphi} \equiv (\varphi_1, \ldots, \varphi_L, 0, \ldots, 0)$ is $L = O(1)$,

$$\langle \phi|\hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi})\hat{G}_0|\psi\rangle \tag{61}$$

can be exactly computed in polynomial time in M and $n_{\max}$.

Using the above lemma, we can immediately see that Lemma 1 is true by using the relationship between $G(\Omega)$ and $\tilde{G}(k)$, which is restated here:

Lemma 4 (Lemma 1 restated). Consider an M-mode bosonic system, product states $|\psi\rangle$, $|\phi\rangle$, a Gaussian unitary $\hat{G}_0$, and an L-mode Fock state $|\boldsymbol{n}\rangle$. Then

$$\langle \phi|\hat{G}_0^\dagger(|\boldsymbol{n}\rangle\langle \boldsymbol{n}| \otimes \hat{\mathbb{1}})\hat{G}_0|\psi\rangle \tag{62}$$

can be exactly computed in polynomial time in M and $n_{\max}$ if $L = O(1)$.

Hence, from now on, we will focus on proving Lemma 3.

Before we do that, we emphasize that the generating function is reduced to the permanent of a matrix with rank depending on L [27,61] when $|\psi\rangle = |\phi\rangle$ are Fock states and the circuit is a linear-optical circuit. We will demonstrate how to compute this quantity as a warm-up example to generalize the approach further below. On the other hand, for Fock states and general Gaussian circuits, the generating function reduces to a loop hafnian [64,90], which also allows a classical algorithm to exploit a low-rank structure to reduce the computational complexity [64]. Notably, such a structure still allows us to compute the generating function even when $|\psi\rangle$ and $|\phi\rangle$ are general product states by using the classical algorithm from Ref. [64], which will be shown in Appendix F.

1. Warm-up example: Permanent

To provide the main technique, we first consider the simplest example: permanent. The permanent of an $N \times N$ matrix A is defined as

$$\mathrm{Per}(A) \equiv \sum_{\sigma \in \mathcal{S}_N} \prod_{i=1}^{N} A_{i,\sigma(i)}. \tag{63}$$

There are many ways to compute the permanent of a matrix, such as Ryser's formula [91]. One of the methods that is relevant to our work is the polynomial method [91]. To be more specific, we define a polynomial of indeterminates $x_1, \ldots, x_N$ as

$$F(x_1, \ldots, x_N) = \prod_{i=1}^{N} \left(\sum_{j=1}^{N} A_{ij} x_j \right). \tag{64}$$

One may then notice that the coefficient of $x_1 \cdots x_N$ corresponds to the permanent of the matrix A.

To illustrate how we leverage the low-rank structure of a relevant matrix, let us review Gurvits' second algorithm [27,61]. Let $A = W + I$, where W is a rank-L matrix, i.e., W's matrix elements can be written as

$$W_{ik} \equiv \sum_{s=1}^{L} u_i^{(s)} v_k^{(s)}. \tag{65}$$

Then, Ref. [61] shows that if we define a polynomial of indeterminates $\{a_u^{(s)}\}_{s=1}^L, \{a_v^{(s)}\}_{s=1}^L$,

$$F(\{a_u^{(s)}\}_{s=1}^L, \{a_v^{(s)}\}_{s=1}^L) = \prod_{i=1}^{N} \left[1 + \sum_{s,s'=1}^{L} a_u^{(s)} a_v^{(s')} u_i^{(s)} v_i^{(s')} \right] \tag{66}$$

and sum the coefficients of monomials whose degrees of $\{a_u^{(s)}\}$ and $\{a_v^{(s)}\}$ match each other with multiplying the factorial of the degrees of $\{a_u^{(s)}\}$ (or equivalently $\{a_v^{(s)}\}$), then we obtain the permanent. Note that the number of monomials that appear in the computation is $O(N^{2L})$. The calculation of the coefficients of the monomials involves multiplying out N terms, where at each multiplication the $O(N^{2L})$ coefficients have to be updated. Therefore, the total cost is $O(N^{2L+1})$. Thus, when the rank of the matrix W is at most a constant, the permanent of the matrix $W + I$ can be efficiently computed, unlike that of general matrices.

2. Amplitude of low-mode states

In the section, for the subroutine of our algorithm for computing the generating function, we consider the amplitude in the form of $\langle\phi|\hat{U}|\psi\rangle$, where $\hat{U}$ is a linear-optical circuit and we assume that $|\phi\rangle$ is an L-mode state, which is defined as

$$|\phi\rangle = |\phi'\rangle \otimes |\mathbf{0}_{M-L}\rangle, \tag{67}$$

where $|\phi'\rangle$ is the nontrivial part and an arbitrary product state on L modes and $|\mathbf{0}_{M-L}\rangle$ is the vacuum state on the remaining $M-L$ modes. We show that the amplitude is easy to compute when L is small:

Lemma 5. Let $\hat{U}$ be an M-mode linear-optical circuit, $|\psi\rangle$ be an M-mode general product state, and $|\phi\rangle$ be an arbitrary L-mode state. More precisely, all but L modes are vacuum. Then,

$$\langle\phi|\hat{U}|\psi\rangle \tag{68}$$

can be exactly computed in $O(M(Mn_{\max}+1)^L)$, where $n_{\max}$ is the maximum photon number for each mode. Hence, it can be efficiently computed when $L = O(1)$.

Proof. Let us first examine Fock states. Consider a few-mode Fock state $|\boldsymbol{n}\rangle = |n_1,\ldots,n_L,0,\ldots,0\rangle$ and arbitrary Fock state $|\boldsymbol{m}\rangle$. Then, the amplitude can be simplified as

$$\langle\boldsymbol{n}|\hat{U}|\boldsymbol{m}\rangle = \langle\boldsymbol{n}|\prod_{i=1}^{M}\frac{1}{\sqrt{m_i!}}\left(\sum_{j=1}^{M}U_{ij}\hat{a}_j^\dagger\right)^{m_i}|0\rangle \tag{69}$$

$$= \langle\boldsymbol{n}|\prod_{i=1}^{M}\frac{1}{\sqrt{m_i!}}\left(\sum_{j=1}^{L}U_{ij}\hat{a}_j^\dagger\right)^{m_i}|0\rangle. \tag{70}$$

Now, note that the creation operators in the form of $(\hat{a}_1^\dagger)^{n_1}\cdots(\hat{a}_L^\dagger)^{n_L}$ survive and others are zero when contracted with $\langle\boldsymbol{n}|$. Also, such a term renders $\prod_{i=1}^{M}\sqrt{n_i}$ after contracting with $\langle\boldsymbol{n}|$. This can also be interpreted using the polynomial method by replacing the creation operators by indeterminates $x_1,\ldots,x_L$ and introducing the prefactor $\prod_{i=1}^{M}\sqrt{n_i}$ as

$$\prod_{i=1}^{M}\frac{\sqrt{n_i!}}{\sqrt{m_i!}}\left(\sum_{j=1}^{L}U_{ij}x_j\right)^{m_i}. \tag{71}$$

We then just need to pick up the monomials of x_j after expansion such that their degree matches with $\boldsymbol{n}$, i.e., $x_1^{n_1}\cdots x_L^{n_L}$ to compute $\langle\boldsymbol{n}|\hat{U}|\boldsymbol{m}\rangle$. In this procedure, the number of monomials we need to take care of is at most $O((MN+1)^L)$, where N is the total photon number. Also, because we need to keep track of the coefficient for M multiplications, the total cost is $O(M(MN+1)^L)$.

Now, let us consider general product states $|\psi\rangle$ and $|\phi\rangle$, which are explicitly written as

$$|\psi\rangle = \bigotimes_{i=1}^{M}\left(\sum_{m_i=0}^{n_{\max}}a_{m_i}^{(i)}|m_i\rangle\right), \quad |\phi\rangle = \bigotimes_{i=1}^{M}\left(\sum_{n_i=0}^{n_{\max}}b_{n_i}^{(i)}|n_i\rangle\right). \tag{72}$$

For a general case, as in the lemma, we can simply use the same method,

$$\langle\phi|\hat{U}|\psi\rangle = \sum_{\boldsymbol{n}}b_{\boldsymbol{n}}^*\langle\boldsymbol{n}|\prod_{i=1}^{M}\sum_{m_i=0}^{n_{\max}}\frac{a_{m_i}^{(i)}}{\sqrt{m_i!}}\left(\sum_{j=1}^{M}U_{ij}\hat{a}_j^\dagger\right)^{m_i}|0\rangle \tag{73}$$

$$= \sum_{\boldsymbol{n}}b_{\boldsymbol{n}}^*\langle\boldsymbol{n}|\prod_{i=1}^{M}\sum_{m_i=0}^{n_{\max}}\frac{a_{m_i}^{(i)}}{\sqrt{m_i!}}\left(\sum_{j=1}^{L}U_{ij}\hat{a}_j^\dagger\right)^{m_i}|0\rangle \tag{74}$$

$$\to \prod_{i=1}^{M}\sum_{n_i,m_i=0}^{n_{\max}}\frac{a_{m_i}^{(i)}b_{n_i}^{(i)*}\sqrt{n_i!}}{\sqrt{m_i!}}\left(\sum_{j=1}^{L}U_{ij}x_j\right)^{m_i}, \tag{75}$$

where $a_{\boldsymbol{m}} \equiv \prod_{i=1}^{M}a_{m_i}^{(i)}$. For the last arrow, we apply the same method for the Fock state above. Thus, again, after expanding the polynomial of indeterminates $x_1,\ldots,x_L$ and taking the summation over the coefficients of all the monomials, we obtain the amplitude $\langle\phi|\hat{U}|\psi\rangle$. Since the number of monomials we need to consider is at most $(Mn_{\max}+1)^L$, the complexity is bounded by $(Mn_{\max}+1)^L$, where $n_{\max}$ is the maximum photon number for each mode. Again, for the update for each multiplication, the total cost becomes $O(M(Mn_{\max}+1)^L)$. ■

3. Low-rank structure

As shown in the permanent case [61], a low-rank structure of a relevant matrix is crucial to reducing the complexity. Now, we reveal the low-rank structure of the Gaussian circuit that is obtained by $\hat{G} = \hat{G}_0^\dagger\hat{P}(\boldsymbol{\varphi})\hat{G}_0$ when $\boldsymbol{\varphi}$ has a few nonzero phases, which is depicted in Fig. 2. To be more precise,

Lemma 6 (Bloch-Messiah decomposition and low-rank structure). Consider a general Gaussian unitary circuit $\hat{G}_0$ and phase-shifter operator $\hat{P}(\boldsymbol{\varphi})$ with $\boldsymbol{\varphi} = (\varphi_1,\ldots,\varphi_L,0,\ldots,0)$. Then, $\hat{G}_0^\dagger\hat{P}(\boldsymbol{\varphi})\hat{G}_0$ can be decomposed into a linear-optical circuit, at most $2L$ nonzero squeezing operators, and another linear-optical circuit.

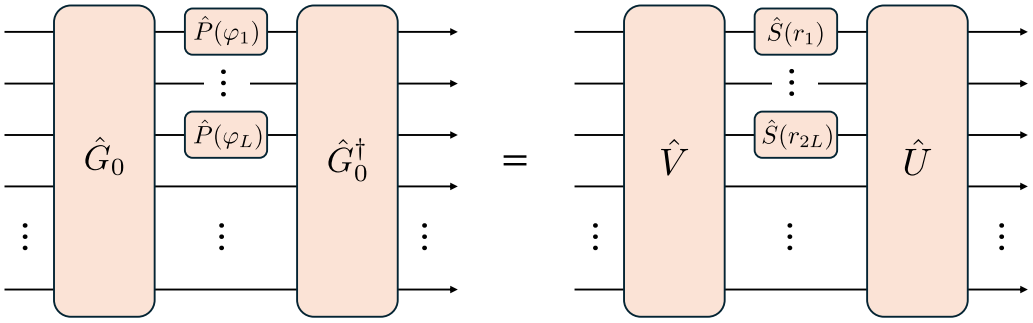


FIG. 2. Schematics of Lemma 6. The lemma shows that for a Gaussian unitary circuit $\hat{G}_0$ and a phase shifter $\hat{P}(\boldsymbol{\varphi})$ where $\boldsymbol{\varphi} \equiv (\varphi_1, \ldots, \varphi_L, 0, \ldots, 0)$, $\hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0$ is equivalent to $\hat{U}\hat{S}(\boldsymbol{r})\hat{V}$, where $\boldsymbol{r} = (r_1, \ldots, r_{2L}, 0, \ldots, 0)$.

Proof sketch. See Appendix C for the details. We prove this lemma by direct calculation of the transformation of bosonic annihilation operators through the Gaussian unitary circuit $\hat{G} = \hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0$, where $\hat{G}_0$ is an M-mode general Gaussian unitary operation, so that using the Bloch-Messiah decomposition it can be written as $\hat{G}_0 = \hat{U}\hat{S}(\boldsymbol{s})\hat{V}$, where $\hat{U}$ and $\hat{V}$ are linear-optical circuits and $\hat{S}(\boldsymbol{s})$ is squeezing operator with squeezing parameter $\boldsymbol{s} \in \mathbb{R}^M$. For this circuit, we can show that

$$\hat{G}_0 \hat{a}_i \hat{G}_0^\dagger = \sum_{k=1}^{M} \left(A_{ik} \hat{a}_k + B_{ik} \hat{a}_k^\dagger \right), \tag{76}$$

where we defined $M \times M$ matrices

$$A_{ik} \equiv \sum_{j=1}^{M} V_{ji}^* \cosh s_j \, U_{kj}^*, \quad B_{ik} \equiv -\sum_{j=1}^{M} V_{ji}^* \sinh s_j \, U_{kj}. \tag{77}$$

Here, one can see that the rank of the matrix B is equal to the number of nonzero squeezing parameters s_j's, which is crucial for later use.

Using the above transformation, by direct calculation, one may easily show that after the entire process $\hat{G}$, we have

$$\hat{G}^\dagger \hat{a}_i \hat{G} = \hat{a}_i + \sum_{k=1}^{M} \left(W_{ik} \hat{a}_k + Z_{ik} \hat{a}_k^\dagger \right), \tag{78}$$

where, if assume that $\varphi_j = 0$ for $j > L$, W and Z's matrix elements are written as

$$W_{ik} = \sum_{j=1}^{L} \left[(e^{i\varphi_j} - 1) A_{ij} A_{kj}^* - (e^{-i\varphi_j} - 1) B_{ij} B_{kj}^* \right], \tag{79}$$

$$Z_{ik} = \sum_{j=1}^{L} \left[-(e^{i\varphi_j} - 1) A_{ij} B_{kj} + (e^{-i\varphi_j} - 1) B_{ij} A_{kj} \right]. \tag{80}$$

Hence, the ranks of W and Z are at most $2L$. Thus, we can write the matrices as

$$W = \sum_{j=1}^{2L} u^{(j)} v^{(j)\mathrm{T}}, \quad Z = \sum_{j=1}^{2L} w^{(j)} z^{(j)\mathrm{T}}. \tag{81}$$

In addition, the low-rankness of the matrix Z implies that the Gaussian transform by $\hat{G} = \hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0$ has a structure in which its squeezing parameters are nonzero only for $2L$ modes, which completes the proof. ■

4. Proof sketch of Lemma 3

In this section, we finally present the sketch of the classical algorithm that computes the generating function for Lemma 3. More specifically, the circuit is a general Gaussian unitary $\hat{G}_0$, and thus the relevant quantity to be computed is written as

$$\langle \phi | \hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0 | \psi \rangle \equiv \langle \phi | \hat{G} | \psi \rangle = \sum_{\boldsymbol{n},\boldsymbol{m}} a_{\boldsymbol{m}} b_{\boldsymbol{n}}^* \langle \boldsymbol{n} | \hat{G} | \boldsymbol{m} \rangle, \tag{82}$$

where $|\psi\rangle$ and $|\phi\rangle$ are arbitrary product states, which are written as Eq. (72). Also, the phase vector is assumed to be written as $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_L, 0, \ldots, 0)$, without loss of generality. By using Lemma 6, the Bloch-Messiah decomposition of the Gaussian unitary $\hat{G}$ is written as $\hat{U}\hat{S}(\boldsymbol{r})\hat{V}$, where $\hat{U}$ and $\hat{V}$ are linear-optical unitary circuits, and $\boldsymbol{r}$ is the squeezing parameter vector with at most $2L$ nonzero elements.

Let us focus on a simpler case: $\langle \boldsymbol{n} | \hat{G} | \boldsymbol{m} \rangle$. Using the low-rank structure, it can be rewritten as

$$\langle \boldsymbol{n} | \hat{G} | \boldsymbol{m} \rangle = \frac{1}{\sqrt{\boldsymbol{n}!}} \langle \boldsymbol{r} | \hat{V} \left[\prod_{i \in [M]} \left(\hat{A}_i + \hat{B}_i^\dagger \right)^{n_i} \right] | \boldsymbol{m} \rangle, \tag{83}$$

where we defined

$$\hat{A}_i = \hat{a}_i + \sum_{k=1}^{M} W_{ik} \hat{a}_k, \quad \hat{B}_i^\dagger = \sum_{k=1}^{M} Z_{ik} \hat{a}_k^\dagger, \tag{84}$$

$$C_{ij} = [\hat{A}_i, \hat{B}_j^\dagger] = Z_{ji} + \sum_{k=1}^{M} W_{ik} Z_{jk}. \tag{85}$$

Note that by Lemma 6, the rank of the matrix C is at most $2L$ since W and Z have at most $2L$ ranks and the number of nonzero squeezing parameters is limited to at most $2L$ as shown in Lemma 6.

By using the antinormal ordering [92] (see Appendix E), we expand Eq. (83) and move all the creation operators to the right,

$$\frac{1}{\sqrt{\boldsymbol{n}!\boldsymbol{m}!}} \sum_{Y \subset X_{\boldsymbol{n}}} \sum_{\mathcal{M} \in \mathrm{Match}(F_{X_{\boldsymbol{n}}}; Y \to Y^c)} (-1)^{|\mathcal{M}|} \left(\prod_{(i,j)\in\mathcal{M}} C_{ij} \right) \times \langle \boldsymbol{r}| \hat{V} \left(\prod_{i \in Y \setminus A(\mathcal{M})} \hat{A}_i \right) \left(\prod_{i \in Y^c \setminus B(\mathcal{M})} \hat{B}_i^\dagger \right) \prod_{i \in X_{\boldsymbol{m}}} \hat{a}_i^\dagger |0\rangle, \tag{86}$$

where $X_{\boldsymbol{n}}$ is the multiset composed of the elements of $\boldsymbol{n}$. For example, if $\boldsymbol{n} = (0,2,1)$, $X_{\boldsymbol{n}} = \{2,2,3\}$. $\mathrm{Match}(F_{X_{\boldsymbol{n}}}; Y \to Y^c)$ is the match of the Ferrers graph [93] with vertices Y and Y^c. More explicitly, the Ferrers graph's edges are defined as $E = \{(i,j)|i,j \in X_{\boldsymbol{n}}, i < j\}$. $A(\mathcal{M})$ and $B(\mathcal{M})$ represent the elements in Y and Y^c that are involved in the matching $\mathcal{M}$, respectively.

We then show that the above expression can be computed by using the polynomial method introduced in Sec. VI B 1. More precisely, we show that Eq. (86) can be written as a loop hafnian of a matrix K which has a low-rank structure. After that, since the matrix C has a low rank, which turns out to be relevant to the low-rank structure of the matrix K, we invoke the following lemma, which states that a loop hafnian of a matrix with a low-rank structure is easy to compute [64]:

Lemma 7 (Loop hafnian of a matrix with low-rank structure [63,64]). Let Σ be an $n \times n$ symmetric matrix with rank r and A be a matrix constructed by filling Σ's diagonal matrix with arbitrary elements. Then the loop hafnian of the matrix A can be computed in $O(n\binom{2n+r-1}{r-1})$. Thus, when r is fixed, the cost is polynomial in n and when $r = O(\log n)$, the cost is quasipolynomial in n.

Another observation to make this calculation easier is that the number of nonzero squeezing parameters is at most $2L$, so that the term $\langle \boldsymbol{r}| \cdots |0\rangle$ in Eq. (86) can be efficiently computed using Lemma 5. In short, we write Eq. (86) as a polynomial of indeterminates, the number of which is at most $(Mn_{\max} + 1)^{16L}$, which makes it easy to compute when $L = O(1)$. We note that while the degree of the polynomial is quite large at this moment, we expect that it may be reduced, which we leave open. Also, if we restrict our attention to linear-optical circuits, the degree of the polynomial reduces to $(Mn_{\max} + 1)^{2L}$ (see Appendix D).

VII. CONCLUSION

In this work, we presented classical algorithms for solving the quantum mean-value problem in Gaussian circuits, as well as in Gaussian circuits augmented by (i) photon-number-resolving measurements with Gaussian feedforward and (ii) Gaussian measurements with Gaussian feedforward. To support these results, we also developed algorithms for computing amplitudes of low-mode states and marginal probabilities of number-basis measurements, which may be of independent interest and connect to Gurvits' second algorithm.

Our results reveal several conceptual insights. First, low-mode and low-rank structures play a central role in making certain bosonic computations classically tractable. Second, we highlight an important contrast between sampling problems and the quantum mean-value problems: while non-Gaussian resources typically induce classical hardness in sampling, mean-value estimation can remain efficiently simulatable even in the presence of highly non-Gaussian inputs, provided the number of adaptive measurements is bounded. This distinction illustrates a richer landscape of classical–quantum complexity for bosonic systems.

Taken together, these findings not only provide concrete classical algorithms but also clarify where the boundaries of quantum advantage may lie in Gaussian and non-Gaussian regimes. We anticipate that these tools and perspectives will aid future studies of bosonic circuits, both as a benchmark for quantum experiments and as a stepping stone toward understanding the computational power of non-Gaussian resources.

On the other hand, since our results emphasize the role of adaptive measurements in the quantum mean-value problem, it is natural to ask what role adaptive measurements play in the corresponding sampling tasks. In our setting, increasing the amount of adaptivity leads to an easy-to-hard transition in the complexity of the mean-value problem, whereas in the nonadaptive regime sampling is already believed to be hard due to boson sampling. It is therefore unclear whether, and in what sense, additional rounds of adaptive (or non-Gaussian) operations change the complexity of the sampling problem. There are related studies that add nonlinear operations to boson sampling [40], but neither our results nor these works currently provide a satisfactory complexity-theoretic understanding of how sampling interpolates between nonuniversal boson sampling and universal photonic quantum computation.

While we generalized the classical algorithm for quantum mean-value problem in Ref. [77], the latter provides another classical algorithm that computes the transition amplitude of a quantum state under linear-optical circuits. We believe that it is an interesting open question to extend this algorithm to the adaptive setting due to its close relation to various applications of bosonic circuits (e.g., quantum kernel methods [44,45]).

Finally, we note that our generalized Gurvits' second algorithm takes polynomial time in the number of modes M when the number of measurements for feedforward is at most constant, i.e., $L = O(1)$, and the degree of the polynomial is not high for linear-optical circuits, as discussed above. Hence, our algorithm is expected to be practically efficient to simulate recent experiments, e.g., the one in Ref. [45]. On the other hand, while our generalized Gurvits' second algorithm for general Gaussian circuits also takes polynomial time when $L = O(1)$, the degree of the polynomial is quite high, which may make it less practical. Hence, improving the practical running time is an important open question. More importantly, whether our algorithm can be improved further to be efficient even when L is logarithmic in the system size is another open question.

ACKNOWLEDGMENTS

C.O. and Y.L. were supported by the National Research Foundation of Korea Grants (No. RS-2023-NR119931, No. RS-2024-00431768, and No. RS-2025-00515456) funded by the Korean government [Ministry of Science and ICT (MSIT)] and the Institute of Information & Communications Technology Planning & Evaluation (IITP) Grants funded by the Korea government (MSIT) (No. RS-2024-00437284, No. IITP-2025-RS-2025-02283189, and No. IITP-2025-RS-2025-02263264).

DATA AVAILABILITY

No data were created or analyzed in this study.

APPENDIX A: QUANTUM MEAN-VALUE PROBLEM PROOFS

In this Appendix, we provide detailed proofs for classical algorithms for quantum mean-value problems.

1. Quantum mean-value problem in Gaussian circuits without feedforward

We provide the full proof of Theorem 1, which is restated here:

Theorem A1 (Theorem 1 restated). Consider an M-mode Gaussian circuit $\hat{G}$, an arbitrary product input state $|\psi\rangle$, and an M-mode product operator $\hat{O} = \hat{O}_A \otimes \hat{\mathbb{1}}_B$. The expectation value $\langle\psi|\hat{G}^\dagger\hat{O}\hat{G}|\psi\rangle$ can be approximated within additive error ϵ with probability $1-\delta$ in running time

$$O\left(\frac{M^2\|\hat{O}_A\|_2^2\mathrm{Tr}[\hat{\rho}_A^2]\log(1/\delta)}{\epsilon^2}\right), \tag{A1}$$

where $\hat{\rho}_A \equiv \mathrm{Tr}_B[\hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger]$ is the reduced density matrix on the system A.

Proof. Let us consider a general Gaussian unitary circuit $\hat{G}$ and a general product input state $|\psi\rangle$. Then, the expectation value of a product observable $\hat{O} = \hat{O}_A \otimes \hat{\mathbb{1}}_B$ can be rewritten as

$$\begin{aligned}
&\langle\psi|\hat{G}^\dagger(\hat{O}_A\otimes\hat{\mathbb{1}}_B)\hat{G}|\psi\rangle \\
&\quad= \frac{1}{\pi^l}\int d^{2l}\boldsymbol{\alpha}_A \mathrm{Tr}[\hat{O}_A\hat{D}(\boldsymbol{\alpha}_A)]\mathrm{Tr}[(\hat{D}^\dagger(\boldsymbol{\alpha}_A)\otimes\hat{\mathbb{1}}_B) \\
&\qquad\times \hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger] &\text{(A2)}\\
&\quad= \frac{1}{\pi^l}\int d^{2l}\boldsymbol{\alpha}_A \mathrm{Tr}[\hat{O}_A\hat{D}(\boldsymbol{\alpha}_A)]\mathrm{Tr}[\hat{D}^\dagger(\boldsymbol{\alpha}_A)\hat{\rho}_A] &\text{(A3)}\\
&\quad= \frac{1}{\pi^l}\int d^{2l}\boldsymbol{\alpha}_A \chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A) &\text{(A4)}\\
&\quad= \int d^{2l}\boldsymbol{\alpha}_A \frac{|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)|^2}{\pi^l\|\hat{O}_A\|_2^2}\frac{\chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A)\|\hat{O}_A\|_2^2}{\chi^*_{\hat{O}_A}(\boldsymbol{\alpha})_A} &\text{(A5)}\\
&\quad\equiv \int d^{2l}\boldsymbol{\alpha}_A p(\boldsymbol{\alpha}_A)X(\boldsymbol{\alpha}_A), &\text{(A6)}
\end{aligned}$$

where we defined the reduced density matrix, a probability distribution, and a random variable,

$$\hat{\rho}_A \equiv \mathrm{Tr}_B[\hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger], \quad p(\boldsymbol{\alpha}_A) = \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l\|\hat{O}_A\|_2^2},$$
$$X(\boldsymbol{\alpha}_A) = \frac{\chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A)\|\hat{O}_A\|_2^2}{\chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)}. \tag{A7}$$

Here, using the property of the displacement operator, we have

$$\frac{1}{\pi^l}\int d^{2l}\boldsymbol{\alpha}_A|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)|^2 = \mathrm{Tr}[\hat{O}_A^\dagger\hat{O}_A] = \|\hat{O}_A\|_2^2, \tag{A8}$$

which shows that $p(\boldsymbol{\alpha}_A)$ is a proper probability distribution. Thus, the above expression allows us to use the median-of-means estimator for estimating the mean value.

We now analyze the complexity. First of all, sampling $\boldsymbol{\alpha}_A$ from $p(\boldsymbol{\alpha}_A)$ is easy due to the product structure of $\hat{O}_A$,

$$p(\boldsymbol{\alpha}_A) = \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l\|\hat{O}_A\|_2^2} = \prod_{i\in A}\frac{\left|\chi_{\hat{O}_i}(\alpha_i)\right|^2}{\pi\|\hat{O}_i\|_2^2}, \tag{A9}$$

which takes $O(l)$ time to compute. Also, the random variable $X(\boldsymbol{\alpha}_A)$ is easy to compute because $\|\hat{O}_A\|_2^2$ is easy to compute due to the product structure and $\chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)$ is easy

to compute as shown above and

$$\begin{aligned}\chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A) &= \mathrm{Tr}[\hat{D}^\dagger(\boldsymbol{\alpha}_A)\hat{\rho}_A] \\ &= \mathrm{Tr}[(\hat{D}^\dagger(\boldsymbol{\alpha}_A)\otimes\hat{\mathbb{1}}_B)\hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger] \\ &= \langle\psi|\hat{D}^\dagger(\boldsymbol{\alpha}')|\psi\rangle = \prod_{i=1}^{M}\langle\psi_i|\hat{D}^\dagger(\alpha'_i)|\psi_i\rangle, \quad \text{(A10)}\end{aligned}$$

where we used the property of displacement operators and Gaussian unitary circuits,

$$\hat{D}^\dagger(\boldsymbol{\alpha}') = \hat{G}^\dagger(\hat{D}^\dagger(\boldsymbol{\alpha}_A)\otimes\hat{\mathbb{1}}_B)\hat{G}. \quad \text{(A11)}$$

Here, $\boldsymbol{\alpha}'$ has a linear relation with $\boldsymbol{\alpha}_A$ so that computing $\boldsymbol{\alpha}'$ takes at most $O(lM)$ time. Therefore, the random variable can be rewritten as

$$\begin{aligned}X(\boldsymbol{\alpha}_A) &= \frac{\chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A)\|\hat{O}_A\|_2^2}{\chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)} = \chi^*_{\hat{\rho}_A}(\boldsymbol{\alpha}_A)\prod_{i\in A}\frac{\|\hat{O}_i\|_2^2}{\chi^*_{\hat{O}_i}(\alpha_i)} \\ &= \mathrm{Tr}[\hat{G}|\psi\rangle\langle\psi|\hat{G}^\dagger\hat{D}(\boldsymbol{\alpha}_A)]^*\prod_{i\in A}\frac{\|\hat{O}_i\|_2^2}{\chi^*_{\hat{O}_i}(\alpha_i)} \quad \text{(A12)} \\ &= \mathrm{Tr}[|\psi\rangle\langle\psi|\hat{D}(\boldsymbol{\alpha}'_A)]^*\prod_{i\in A}\frac{\|\hat{O}_i\|_2^2}{\chi^*_{\hat{O}_i}(\alpha_i)} \\ &= \prod_{i=1}^{M}\chi^*_{\psi_i}(\alpha'_i)\prod_{i\in A}\frac{\|\hat{O}_i\|_2^2}{\chi^*_{\hat{O}_i}(\alpha_i)}, \quad \text{(A13)}\end{aligned}$$

and computing this takes at most $O(l+M)$ cost.

Finally, we find the upper bound on the variance of the random variable,

$$\begin{aligned}\mathrm{Var}(X) &\le \int d^{2l}\boldsymbol{\alpha}_A p(\boldsymbol{\alpha}_A)|X(\boldsymbol{\alpha}_A)|^2 \\ &= \int d^{2l}\boldsymbol{\alpha}_A \frac{|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)|^2}{\pi^l\|\hat{O}_A\|_2^2}\frac{|\chi_\psi(\boldsymbol{\alpha}')|^2\|\hat{O}_A\|_2^4}{|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)|^2} \\ &= \|\hat{O}_A\|_2^2\int d^{2l}\boldsymbol{\alpha}_A\frac{|\chi_\psi(\boldsymbol{\alpha}')|^2}{\pi^l} \quad \text{(A14)} \\ &= \|\hat{O}_A\|_2^2\|\hat{\rho}_A\|_2^2, \quad \text{(A15)}\end{aligned}$$

where, for the last equality, we used

$$\begin{aligned}&\int d^{2l}\boldsymbol{\alpha}_A\frac{|\chi_\psi(\boldsymbol{\alpha}')|^2}{\pi^l} \\ &= \int d^{2l}\boldsymbol{\alpha}_A\frac{|\langle\psi|\hat{G}^\dagger(\hat{D}^\dagger(\boldsymbol{\alpha}_A)\otimes\hat{\mathbb{1}}_B)\hat{G}|\psi\rangle|^2}{\pi^l} \\ &= \int d^{2l}\boldsymbol{\alpha}_A\frac{|\chi_{\hat{\rho}_A}(\boldsymbol{\alpha}_A)|^2}{\pi^l} = \|\hat{\rho}_A\|_2^2. \quad \text{(A16)}\end{aligned}$$

Hence, the median-of-means estimator can estimate the mean value with the sampling complexity $O(\|\hat{O}_A\|_2^2\|\hat{\rho}_A\|_2^2\log(1/\delta)/\epsilon^2)$. Thus, combining all the complexities, the total running time is given as $O(Ml\|\hat{O}_A\|_2^2\|\hat{\rho}_A\|_2^2\log(1/\delta)/\epsilon^2)$. ■

2. Quantum mean-value problem in Gaussian circuits with photon-number measurements and Gaussian feedforward

We provide the full proof of Theorem 3, which is restated here:

Theorem A2 (Theorem 3 restated). Consider an arbitrary product input state and a Gaussian circuit, augmented by photon-number measurements and Gaussian feedforward, which outputs an M-mode quantum state. If the number of photon-number measurements used for feedforward is $L = O(1)$, then the expectation value of an M-mode observable $\hat{O} = \hat{O}_A\otimes\hat{\mathbb{1}}_B$ can be approximated within additive error ϵ and success probability $1-\delta$ in running time,

$$O\left(\frac{\mathrm{poly}(M)\|\hat{O}_A\|_2^2\mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{n}}^2)]\log(1/\delta)}{\epsilon^2}\right), \quad \text{(A17)}$$

where $\hat{\rho}_{\boldsymbol{n}}$ is the reduced density matrix on the system A when the photon-number measurement outcome is $\boldsymbol{n}$ and $\mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{n}}^2)]$ is the average purity.

Proof. Consider a general product input $|\psi\rangle$. First of all, when the measurement outcomes are $\boldsymbol{n}$, the corresponding postmeasurement output state is (up to normalization) (see Sec. V B)

$$\langle\boldsymbol{n}|\hat{G}_{\boldsymbol{n}}|\psi\rangle, \quad \text{(A18)}$$

with probability

$$\begin{aligned}p(\boldsymbol{n}) &= \langle\psi|\hat{G}_{\boldsymbol{n}}^\dagger(\hat{\mathbb{1}}_M\otimes|\boldsymbol{n}\rangle\langle\boldsymbol{n}|)\hat{G}_{\boldsymbol{n}}|\psi\rangle \\ &= \langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}}_M\otimes|\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle. \quad \text{(A19)}\end{aligned}$$

Thus, the expectation value when the outcome is $\boldsymbol{n}$ is

$$\frac{\langle\psi_{\boldsymbol{n}}|(\hat{O}_A\otimes\hat{\mathbb{1}}_B\otimes|\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle}{\langle\psi_{\boldsymbol{n}}|(\hat{\mathbb{1}}_A\otimes\hat{\mathbb{1}}_B\otimes|\boldsymbol{n}\rangle\langle\boldsymbol{n}|)|\psi_{\boldsymbol{n}}\rangle}. \quad \text{(A20)}$$

Since the measurement $\boldsymbol{n}$ is obtained with probability $p(\boldsymbol{n})$, we take the average over all possible outcomes in the middle,

$$\langle \hat{O} \rangle = \sum_{\boldsymbol{n}} \langle \psi_{\boldsymbol{n}} | (\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle \tag{A21}$$

$$= \frac{1}{\pi^l} \int d^{2l}\boldsymbol{\alpha} \sum_{\boldsymbol{n}} \langle \psi_{\boldsymbol{n}} | (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle \chi_{\hat{O}_A}(\boldsymbol{\alpha}_A) \tag{A22}$$

$$= \int d^{2l}\boldsymbol{\alpha}_A \sum_{\boldsymbol{n}} p(\boldsymbol{n}) q(\boldsymbol{\alpha}_A) \frac{\|\hat{O}_A\|_2^2 \langle \psi_{\boldsymbol{n}} | (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle}{p(\boldsymbol{n}) \chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)} \tag{A23}$$

$$= \int d^{2l}\boldsymbol{\alpha}_A \sum_{\boldsymbol{n}} p(\boldsymbol{n}) q(\boldsymbol{\alpha}_A) X(\boldsymbol{n}, \boldsymbol{\alpha}_A), \tag{A24}$$

where we defined the probability distribution and the random variable

$$q(\boldsymbol{\alpha}_A) \equiv \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l \|\hat{O}_A\|_2^2}, \quad X(\boldsymbol{n}, \boldsymbol{\alpha}_A) \equiv \frac{\|\hat{O}_A\|_2^2 \langle \psi_{\boldsymbol{n}} | (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle}{p(\boldsymbol{n}) \chi^*_{\hat{O}_A}(\boldsymbol{\alpha}_A)}. \tag{A25}$$

Now, we analyze the complexity. First, to upper-bound the error of the estimator, we compute the second moment,

$$\int d^{2l}\boldsymbol{\alpha}_A \sum_{\boldsymbol{n}} p(\boldsymbol{n}) q(\boldsymbol{\alpha}_A) \frac{\|\hat{O}_A\|_2^4 |\langle \psi_{\boldsymbol{n}} | (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle|^2}{p(\boldsymbol{n})^2 \left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2} \tag{A26}$$

$$= \frac{\|\hat{O}_A\|_2^2}{\pi^l} \int d^{2l}\boldsymbol{\alpha}_A \sum_{\boldsymbol{n}} \frac{|\langle \psi_{\boldsymbol{n}} | (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle|^2}{p(\boldsymbol{n})} \tag{A27}$$

$$= \frac{\|\hat{O}_A\|_2^2}{\pi^l} \int d^{2l}\boldsymbol{\alpha}_A \sum_{\boldsymbol{n}} p(\boldsymbol{n}) |\chi_{\hat{\rho}_{\boldsymbol{n}}}(\boldsymbol{\alpha}_A)|^2 \tag{A28}$$

$$= \|\hat{O}_A\|_2^2 \sum_{\boldsymbol{n}} p(\boldsymbol{n}) \mathrm{Tr}[\hat{\rho}_{\boldsymbol{n}}^2], \tag{A29}$$

where $\hat{\rho}_{\boldsymbol{n}}$ is the reduced density matrix on the system A when the measurement outcome is $\boldsymbol{n}$. Thus, the variance is upper-bounded by the Hilbert-Schmidt norm of the operator $\hat{O}_A$ multiplied by the average purity of the reduced state on the observable's Hilbert space. Hence, the sample complexity for the median-of-means estimator is given as $O(\|\hat{O}_A\|_2^2 \mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}[\hat{\rho}_{\boldsymbol{n}}^2]] \log(1/\delta)/\epsilon^2)$.

Meanwhile, we need to be able to sample from $p(\boldsymbol{\beta})$ and $q(\boldsymbol{\alpha}_A)$. Sampling from $q(\boldsymbol{\alpha}_A)$ can be efficiently done because of the product structure,

$$q(\boldsymbol{\alpha}_A) = \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l \|\hat{O}_A\|_2^2} = \prod_{i \in A} \frac{|\chi_{\hat{O}_i}(\alpha_i)|^2}{\pi \|\hat{O}_i\|_2^2}, \tag{A30}$$

which takes at most $O(l)$. Also, $p(\boldsymbol{n})$ can be easily sampled if the number of modes in $|\boldsymbol{n}\rangle$ is not too large by Lemma 1. More specifically, when the number of modes in $|\boldsymbol{n}\rangle$ is $L = O(1)$, sampling from $p(\boldsymbol{n})$ takes at most polynomial in M.

Also, in computing the random variable $X(\boldsymbol{n}, \boldsymbol{\alpha}_A)$, $\|\hat{O}_A\|_2^2 = \prod_{i \in A} \|\hat{O}_i\|_2^2$ is easy to compute, and $p(\boldsymbol{n})$ is the same as the above. Also, $\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)$ is easy to compute due to the product structure of $\hat{O}_A$ [see Eq. (A9)]. Finally,

$$\langle \psi_{\boldsymbol{n}} | (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) | \psi_{\boldsymbol{n}} \rangle$$

$$= \langle \psi | \hat{G}_{\boldsymbol{n}}^\dagger (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) \hat{G}_{\boldsymbol{n}} | \psi \rangle$$

$$= \langle \psi | \hat{D}^\dagger(\boldsymbol{\alpha}') \hat{G}_{\boldsymbol{n}}^\dagger (\hat{\mathbb{1}}_A \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) \hat{G}_{\boldsymbol{n}} | \psi \rangle \tag{A31}$$

$$= \langle \phi | \hat{G}_{\boldsymbol{n}}^\dagger (\hat{\mathbb{1}}_A \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{n}\rangle\langle \boldsymbol{n}|) \hat{G}_{\boldsymbol{n}} | \psi \rangle, \tag{A32}$$

where we used

$$\hat{G}_{\boldsymbol{n}}^\dagger \hat{D}^\dagger(\boldsymbol{\alpha}_A) \hat{G}_{\boldsymbol{n}} = \hat{D}^\dagger(\boldsymbol{\alpha}'). \tag{A33}$$

Here, computing $\boldsymbol{\alpha}'$ takes at most $O(Ml)$ and again by using Lemma 1, we can compute this quantity in polynomial time when $L = O(1)$. Therefore, the total running time is

at most

$$O\left(\frac{\mathrm{poly}(M)\|\hat{O}_A\|_2^2\mathbb{E}_{\boldsymbol{n}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{n}}^2)]\log(1/\delta)}{\epsilon^2}\right), \tag{A34}$$

which completes the proof. ■

3. Quantum mean-value problem in Gaussian circuit augmented by Gaussian measurement and Gaussian feedforward

We provide the full proof of Theorem 4, which is restated here:

Theorem A3 (Theorem 4 restated). Consider an arbitrary product input and a Gaussian circuit, augmented by Gaussian measurements and Gaussian feedforward, which outputs an M-mode quantum state. If the number of Gaussian measurements for feedforward is at most constant, i.e., $L = O(1)$, the expectation value of an M-mode product observable $\hat{O} \equiv \hat{O}_A \otimes \hat{\mathbb{1}}_B$ can be approximated within additive error ϵ with probability $1-\delta$ in running time,

$$O\left(\frac{\mathrm{poly}(M)\|\hat{O}_A\|_2^2\mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{\beta}}^2)]\log(1/\delta)}{\epsilon^2}\right), \tag{A35}$$

where $\hat{\rho}_{\boldsymbol{\beta}}$ is the reduced density matrix on the system A when the Gaussian measurement outcome is $\boldsymbol{\beta}$ and $\mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{\beta}}^2)]$ is the average purity.

Proof. Consider a general product input $|\psi\rangle$ and Gaussian measurement POVMs,

$$\frac{1}{\pi^L}\hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}). \tag{A36}$$

First of all, when the measurement outcomes are $\boldsymbol{\beta}$, the probability of obtaining $\boldsymbol{\beta}$ is

$$\begin{aligned} p(\boldsymbol{\beta}) &= \frac{1}{\pi^L}\langle\psi|\hat{G}_{\boldsymbol{\beta}}^\dagger(\hat{\mathbb{1}}_M \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))\hat{G}_{\boldsymbol{\beta}}|\psi\rangle \\ &= \frac{1}{\pi^L}\langle\psi_{\boldsymbol{\beta}}|(\hat{\mathbb{1}}_M \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle, \end{aligned} \tag{A37}$$

and the (unnormalized) postmeasurement state is

$$\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta})|\psi_{\boldsymbol{\beta}}\rangle. \tag{A38}$$

Thus, the expectation value when the outcome is $\boldsymbol{\beta}$ is

$$\frac{\langle\psi_{\boldsymbol{\beta}}|(\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle}{\langle\psi_{\boldsymbol{\beta}}|(\hat{\mathbb{1}}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{\Pi}(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle}. \tag{A39}$$

Taking the average over all outcomes $\boldsymbol{\beta}$ and defining $|\psi_{\boldsymbol{\beta}}\rangle = \hat{G}_{\boldsymbol{\beta}}|\psi\rangle$, the expectation value is written as

$$\langle\hat{O}\rangle = \frac{1}{\pi^L}\int d^{2L}\boldsymbol{\beta}\langle\psi_{\boldsymbol{\beta}}|(\hat{O}_A \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle \tag{A40}$$

$$= \frac{1}{\pi^L}\int \frac{d^{2l}\boldsymbol{\alpha}_A}{\pi^l}d^{2L}\boldsymbol{\beta}\langle\psi_{\boldsymbol{\beta}}|(\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A) \tag{A41}$$

$$= \frac{1}{\pi^L}\int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta}\frac{\|\hat{O}_A\|_2^2\langle\psi_{\boldsymbol{\beta}}|(\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle}{\chi_{\hat{O}_A}^*(\boldsymbol{\alpha}_A)}\frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l\|\hat{O}_A\|_2^2} \tag{A42}$$

$$= \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta}p(\boldsymbol{\beta})\frac{\|\hat{O}\|_2^2\langle\psi_{\boldsymbol{\beta}}|(\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle}{\pi^L p(\boldsymbol{\beta})\chi_{\hat{O}_A}^*(\boldsymbol{\alpha}_A)}q(\boldsymbol{\alpha}_A) \tag{A43}$$

$$= \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta}p(\boldsymbol{\beta})q(\boldsymbol{\alpha}_A)X(\boldsymbol{\beta},\boldsymbol{\alpha}_A), \tag{A44}$$

where we defined a probability distribution and a random variable as

$$q(\boldsymbol{\alpha}_A) \equiv \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l\|\hat{O}_A\|_2^2}, \quad X(\boldsymbol{\beta},\boldsymbol{\alpha}) = \frac{\|\hat{O}_A\|_2^2\langle\psi_{\boldsymbol{\beta}}|(\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta})|\psi_G\rangle\langle\psi_G|\hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle}{\pi^L p(\boldsymbol{\beta})\chi_{\hat{O}_A}^*(\boldsymbol{\alpha}_A)}. \tag{A45}$$

Hence, the quantum mean value can be written as

$$\langle \hat{O} \rangle = \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta} p(\boldsymbol{\beta}) q(\boldsymbol{\alpha}_A) X(\boldsymbol{\beta}, \boldsymbol{\alpha}_A), \tag{A46}$$

which implies that the average of the random variable $X(\boldsymbol{\beta}, \boldsymbol{\alpha}_A)$ over the probability distribution $p(\boldsymbol{\beta})q(\boldsymbol{\alpha}_A)$ is equal to the mean value. If we use the median-of-means estimator, we can upper-bound the error of the estimator by finding the upper bound on the second moment:

$$\int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta} p(\boldsymbol{\beta}) q(\boldsymbol{\alpha}_A) \frac{\|\hat{O}_A\|_2^4 |\langle \psi_{\boldsymbol{\beta}}| (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle|^2}{\pi^{2L} p(\boldsymbol{\beta})^2 \left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2} \tag{A47}$$

$$= \frac{\|\hat{O}_A\|_2^2}{\pi^{2L}\pi^l} \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta} \frac{|\langle \psi_{\boldsymbol{\beta}}| (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle|^2}{p(\boldsymbol{\beta})} \tag{A48}$$

$$= \frac{\|\hat{O}_A\|_2^2}{\pi^l} \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta} \frac{|\langle \psi_{\boldsymbol{\beta}}| (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle|^2}{\langle \psi_{\boldsymbol{\beta}}| (\hat{\mathbb{1}}_A \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle} \tag{A49}$$

$$= \frac{\|\hat{O}_A\|_2^2}{\pi^l} \int d^{2l}\boldsymbol{\alpha}_A d^{2L}\boldsymbol{\beta} p(\boldsymbol{\beta}) |\chi_{\hat{\rho}_{\boldsymbol{\beta}}}(\boldsymbol{\alpha}_A)|^2 \tag{A50}$$

$$= \|\hat{O}_A\|_2^2 \int d^{2L}\boldsymbol{\beta} p(\boldsymbol{\beta}) \mathrm{Tr}[\hat{\rho}_{\boldsymbol{\beta}}^2]. \tag{A51}$$

Thus, the variance is upper-bounded by the Hilbert-Schmidt norm of the operator $\hat{O}_A$ multiplied by the average purity of the reduced state when tracing out the observable's Hilbert space. Thus, the sample complexity for the estimator is given as $O(\|\hat{O}_A\|_2^2 \mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}[\hat{\rho}_{\boldsymbol{\beta}}^2]] \log(1/\delta)/\epsilon^2)$.

Meanwhile, we need to be able to sample from $p(\boldsymbol{\beta})$ and $q(\boldsymbol{\alpha}_A)$. Sampling from $q(\boldsymbol{\alpha}_A)$ can be efficiently done because of the product structure,

$$q(\boldsymbol{\alpha}_A) = \frac{\left|\chi_{\hat{O}_A}(\boldsymbol{\alpha}_A)\right|^2}{\pi^l \|\hat{O}_A\|_2^2} = \prod_{i \in A} \frac{|\chi_{\hat{O}_i}(\alpha_i)|^2}{\pi \|\hat{O}_i\|_2^2}, \tag{A52}$$

which takes at most $O(M)$. Also, $p(\boldsymbol{\beta})$ can also be easily sampled if the number of modes L in $|\boldsymbol{\beta}\rangle$ is not too large,

$$p(\boldsymbol{\beta}) = \frac{1}{\pi^L} \langle \psi | \hat{G}_{\boldsymbol{\beta}}^\dagger (\hat{\mathbb{1}}_M \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta})) \hat{G}_{\boldsymbol{\beta}} |\psi\rangle = \frac{1}{\pi^L} \langle \psi | \hat{G}_{\boldsymbol{\beta}}^\dagger \hat{D}^\dagger(\boldsymbol{\beta}) (\hat{\mathbb{1}}_M \otimes |\mathbf{0}\rangle\langle\mathbf{0}|) \hat{D}(\boldsymbol{\beta}) \hat{G}_{\boldsymbol{\beta}} |\psi\rangle \tag{A53}$$

$$= \frac{1}{\pi^L} \langle \psi | \hat{D}^\dagger(\boldsymbol{\beta}') \hat{G}_{\boldsymbol{\beta}}^\dagger (\hat{\mathbb{1}}_M \otimes |\mathbf{0}\rangle\langle\mathbf{0}|) \hat{G}_{\boldsymbol{\beta}} \hat{D}(\boldsymbol{\beta}') |\psi\rangle = \frac{1}{\pi^L} \langle \psi' | \hat{G}_{\boldsymbol{\beta}}^\dagger (\hat{\mathbb{1}}_M \otimes |\mathbf{0}\rangle\langle\mathbf{0}|) \hat{G}_{\boldsymbol{\beta}} |\psi'\rangle, \tag{A54}$$

where we used the property of the displacement operator,

$$\hat{D}^\dagger(\boldsymbol{\alpha}') = \hat{G}_{\boldsymbol{\beta}}^\dagger \hat{D}^\dagger(\boldsymbol{\alpha}) \hat{G}_{\boldsymbol{\beta}}. \tag{A55}$$

More specifically, when $L = O(1)$, by Lemma 1, we can sample from $p(\boldsymbol{\beta})$ in polynomial time.

Finally, we analyze the complexity of computing the random variable X,

$$X(\boldsymbol{\beta}, \boldsymbol{\alpha}) = \frac{\|\hat{O}_A\|_2^2 \langle \psi_{\boldsymbol{\beta}}| (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes |\boldsymbol{\beta}\rangle\langle\boldsymbol{\beta}|) |\psi_{\boldsymbol{\beta}}\rangle}{\pi^L p(\boldsymbol{\beta}) \chi_{\hat{O}_A}^*(\boldsymbol{\alpha}_A)}. \tag{A56}$$

The difficult part in computing the random variable is

$$\langle \psi_{\boldsymbol{\beta}}| (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta}))|\psi_{\boldsymbol{\beta}}\rangle = \langle \psi | \hat{G}^\dagger(\boldsymbol{\beta}) (\hat{D}^\dagger(\boldsymbol{\alpha}_A) \otimes \hat{\mathbb{1}}_B \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta})) \hat{G}(\boldsymbol{\beta}) |\psi\rangle \tag{A57}$$

$$= \langle \psi | \hat{G}^\dagger(\boldsymbol{\beta}) (\hat{\mathbb{1}} \otimes \hat{D}(\boldsymbol{\beta}) |\psi_G\rangle\langle\psi_G| \hat{D}^\dagger(\boldsymbol{\beta})) \hat{G}(\boldsymbol{\beta}) \hat{D}(\boldsymbol{\alpha}_A') |\psi\rangle = \langle \psi | \hat{G}^\dagger(\boldsymbol{\beta}) (\hat{\mathbb{1}} \otimes |\mathbf{0}\rangle\langle\mathbf{0}|) \hat{G}(\boldsymbol{\beta}) |\psi'\rangle, \tag{A58}$$

but this can also be computed by using Lemma 1 when $L = O(1)$. Thus, the total running time is given as

$$O\left(\frac{\mathrm{poly}(M)\|\hat{O}_A\|_2^2\mathbb{E}_{\boldsymbol{\beta}}[\mathrm{Tr}(\hat{\rho}_{\boldsymbol{\beta}}^2)]\log(1/\delta)}{\epsilon^2}\right), \tag{A59}$$

which completes the proof. ■

APPENDIX B: DERIVATION OF $\tilde{G}(k)$

In this Appendix, we provide a derivation of the following relationship between $\tilde{G}(k)$ and $G(\Omega)$ in Eqs. (55) and (56) [64]:

$$\tilde{G}(k) \equiv \langle\phi|\hat{G}_0^\dagger e^{-ik\theta\hat{\boldsymbol{n}}\cdot\boldsymbol{\omega}}\hat{G}_0|\psi\rangle, \tag{B1}$$

$$G(\Omega) \equiv \sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty} q(\boldsymbol{n})\delta(\Omega - \boldsymbol{\omega}\cdot\boldsymbol{n}) = \frac{1}{\Omega_{\max}+1}\sum_{k=0}^{\Omega_{\max}}\tilde{G}(k)e^{ik\Omega}, \tag{B2}$$

where $\theta = 2\pi/(\Omega_{\max}+1)$ and

$$q(\boldsymbol{n}) \equiv \langle\phi|\hat{G}_0^\dagger|\boldsymbol{n}\rangle\langle\boldsymbol{n}|\hat{G}_0|\psi\rangle. \tag{B3}$$

Let us compute the Fourier transform of $G(\Omega)$,

$$\sum_{\Omega=0}^{\Omega_{\max}} G(\Omega)e^{-ik\theta\Omega} = \sum_{\Omega=0}^{\Omega_{\max}}\sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty} q(\boldsymbol{n})\delta(\Omega-\boldsymbol{\omega}\cdot\boldsymbol{n})e^{-ik\theta\Omega} = \frac{1}{\Omega_{\max}+1}\sum_{\Omega=0}^{\Omega_{\max}}\sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty} q(\boldsymbol{n})\sum_{l=0}^{\Omega_{\max}} e^{il\theta(\Omega-\boldsymbol{\omega}\cdot\boldsymbol{n})-ik\theta\Omega} \tag{B4}$$

$$= \frac{1}{\Omega_{\max}+1}\sum_{l=0}^{\Omega_{\max}}\sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty} q(\boldsymbol{n})e^{-il\theta\boldsymbol{\omega}\cdot\boldsymbol{n}}\sum_{\Omega=0}^{\Omega_{\max}} e^{i(l-k)\theta\Omega} = \sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty} q(\boldsymbol{n})e^{-ik\theta\boldsymbol{\omega}\cdot\boldsymbol{n}} = \sum_{\boldsymbol{n}=\boldsymbol{0}}^{\infty}\langle\phi|\hat{G}_0^\dagger|\boldsymbol{n}\rangle\langle\boldsymbol{n}|\hat{G}_0|\psi\rangle e^{-ik\theta\boldsymbol{\omega}\cdot\boldsymbol{n}} \tag{B5}$$

$$= \langle\phi|\hat{G}_0^\dagger e^{-ik\theta\hat{\boldsymbol{n}}\cdot\boldsymbol{\omega}}\hat{G}_0|\psi\rangle = \tilde{G}(k). \tag{B6}$$

Hence, $\tilde{G}(k)$ is the Fourier transformation of $G(\Omega)$, which enables us to compute the marginal probabilities of q.

APPENDIX C: LOW-RANK STRUCTURE IN GAUSSIAN CIRCUITS

In this Appendix, we provide an explicit derivation of the transformation of bosonic annihilation operators through the Gaussian unitary circuit $\hat{G}_0^\dagger\hat{P}(\boldsymbol{\varphi})\hat{G}_0$. To be more specific, we prove Lemma 6:

Lemma C1 (Lemma 6 restated). Consider a general Gaussian unitary circuit $\hat{G}_0$ and phase-shifter operator $\hat{P}(\boldsymbol{\varphi})$ with $\boldsymbol{\varphi} = (\varphi_1,\ldots,\varphi_L,0,\ldots,0)$. Then, $\hat{G}_0^\dagger\hat{P}(\boldsymbol{\varphi})\hat{G}_0$ can be decomposed into a linear-optical circuit, at most $2L$ nonzero squeezing operators, and another linear-optical circuit.

Proof. First of all, the bosonic annihilation operator transforms under $\hat{G}_0 = \hat{U}\hat{S}(\boldsymbol{s})\hat{V}$ as

$$\hat{G}_0\hat{a}_i\hat{G}_0^\dagger = \hat{U}\hat{S}\hat{V}\hat{a}_i\hat{V}^\dagger\hat{S}^\dagger\hat{U}^\dagger = \hat{U}\hat{S}\left(\sum_{j=1}^{M}V_{ji}^*\hat{a}_j\right)\hat{S}^\dagger\hat{U}^\dagger = \hat{U}\left(\sum_{j=1}^{M}V_{ji}^*(\cosh s_j\,\hat{a}_j - \sinh s_j\,\hat{a}_j^\dagger)\right)\hat{U}^\dagger \tag{C1}$$

$$= \sum_{j,k=1}^{M}V_{ji}^*(\cosh s_j\,U_{kj}^*\hat{a}_k - \sinh s_j\,U_{kj}\hat{a}_k^\dagger) = \sum_{k=1}^{M}\left(A_{ik}\hat{a}_k + B_{ik}\hat{a}_k^\dagger\right), \tag{C2}$$

where we used

$$\hat{U}^\dagger \hat{a}_i \hat{U} = \sum_{j=1}^{M} U_{ij}\,\hat{a}_j\,, \quad \hat{U}\hat{a}_i\hat{U}^\dagger = \sum_{j=1}^{M} U^*_{ji}\hat{a}_j\,, \quad \hat{S}^\dagger(s)\hat{a}_i\hat{S}(s) = \cosh s\hat{a}_i + \sinh s\hat{a}_i^\dagger, \quad \hat{S}(s)\hat{a}_i\hat{S}^\dagger(s) = \cosh s\hat{a}_i - \sinh s\hat{a}_i^\dagger, \tag{C3}$$

and we defined

$$A_{ik} \equiv \sum_{j=1}^{M} V^*_{ji}\cosh s_j\, U^*_{kj}\,, \quad B_{ik} \equiv -\sum_{j=1}^{M} V^*_{ji}\sinh s_j\, U_{kj}\,, \tag{C4}$$

which characterize the Gaussian transformation. Here, we emphasize that the number of nonzero squeezing parameters decides the rank of the matrix B. More precisely, if the number of nonzero squeezing parameters is L, then the rank of the matrix B is at most L.

For the transformation under $\hat{G}_0^\dagger$, we similarly have

$$\hat{G}_0^\dagger \hat{a}_i \hat{G}_0 = \hat{V}^\dagger\hat{S}^\dagger\hat{U}^\dagger\hat{a}_i\hat{U}\hat{S}\hat{V} = \hat{V}^\dagger\hat{S}^\dagger\left(\sum_{j=1}^{M} U_{ij}\,\hat{a}_j\right)\hat{S}\hat{V} = \hat{V}^\dagger\left(\sum_{j=1}^{M} U_{ij}\,(\cosh s_j\,\hat{a}_j + \sinh s_j\,\hat{a}_j^\dagger)\right)\hat{V} \tag{C5}$$

$$= \sum_{j,k=1}^{M} U_{ij}\,(\cosh s_j\, V_{jk}\hat{a}_k + \sinh s_j\, V^*_{jk}\hat{a}_k^\dagger) = \sum_{k=1}^{M}\left((A^\dagger)_{ik}\hat{a}_k - (B^{\mathrm{T}})_{ik}\hat{a}_k^\dagger\right). \tag{C6}$$

Because of the unitarity, after the transformation under $\hat{G}_0$ and then $\hat{G}_0^\dagger$, i.e., the identity operation,

$$\hat{a}_i \to \sum_{j=1}^{M} A_{ij}\,\hat{a}_j + \sum_{j=1}^{M} B_{ij}\,\hat{a}_j^\dagger \to \sum_{j,k=1}^{M} A_{ij}\left((A^\dagger)_{jk}\hat{a}_k - (B^{\mathrm{T}})_{jk}\hat{a}_k^\dagger\right) + \sum_{j,k=1}^{M} B_{ij}\left((A^\dagger)^*_{jk}\hat{a}_k^\dagger - (B^{\mathrm{T}})^*_{jk}\hat{a}_k\right) \tag{C7}$$

$$= \sum_{j,k=1}^{M}\left[\left(A_{ij}\,(A^\dagger)_{jk} - B_{ij}\,(B^\dagger)_{jk}\right)\hat{a}_k + \left(-A_{ij}\,(B^{\mathrm{T}})_{jk} + B_{ij}\,(A^{\mathrm{T}})_{jk}\right)\hat{a}_k^\dagger\right], \tag{C8}$$

the matrices A and B must satisfy the following conditions:

$$AA^\dagger - BB^\dagger = \mathbb{1}, \quad AB^{\mathrm{T}} = BA^{\mathrm{T}}. \tag{C9}$$

Let $C \equiv A^\dagger$ and $D = -B^{\mathrm{T}}$. After the entire process, including the phase shifter $\hat{P}(\boldsymbol{\varphi})$, the annihilation operator transforms as

$$\hat{a}_i \to \sum_{j=1}^{M} A_{ij}\,\hat{a}_j + \sum_{j=1}^{M} B_{ij}\,\hat{a}_j^\dagger \to \sum_{j=1}^{M} e^{i\varphi_j} A_{ij}\,\hat{a}_j + \sum_{j=1}^{M} e^{-i\varphi_j} B_{ij}\,\hat{a}_j^\dagger \tag{C10}$$

$$\to \sum_{j,k=1}^{M} e^{i\varphi_j} A_{ij}\left(C_{jk}\hat{a}_k + D_{jk}\hat{a}_k^\dagger\right) + \sum_{j,k=1}^{M} e^{-i\varphi_j} B_{ij}\left(C^*_{jk}\hat{a}_k^\dagger + D^*_{jk}\hat{a}_k\right) \tag{C11}$$

$$= \sum_{j,k=1}^{M} e^{i\varphi_j} A_{ij}\, C_{jk}\hat{a}_k + \sum_{j,k=1}^{M} e^{i\varphi_j} A_{ij}\, D_{jk}\hat{a}_k^\dagger + \sum_{j,k=1}^{M} e^{-i\varphi_j} B_{ij}\, C^*_{jk}\hat{a}_k^\dagger + \sum_{j,k=1}^{M} e^{-i\varphi_j} B_{ij}\, D^*_{jk}\hat{a}_k. \tag{C12}$$

Here, by assuming that $\varphi_j = 0$ for $j > L$, without loss of generality, and using Eq. (C9), we can simplify the expression as

$$\sum_{j,k=1}^{M} e^{i\varphi_j} A_{ij} C_{jk} \hat{a}_k + \sum_{j,k=1}^{M} e^{i\varphi_j} A_{ij} D_{jk} \hat{a}_k^\dagger + \sum_{j,k=1}^{M} e^{-i\varphi_j} B_{ij} C_{jk}^* \hat{a}_k^\dagger + \sum_{j,k=1}^{M} e^{-i\varphi_j} B_{ij} D_{jk}^* \hat{a}_k \tag{C13}$$

$$= \sum_{j=1}^{M}\sum_{k=1}^{M} \left(A_{ij} C_{jk} \hat{a}_k + B_{ij} D_{jk}^* \hat{a}_k + A_{ij} D_{jk} \hat{a}_k^\dagger + B_{ij} C_{jk}^* \hat{a}_k^\dagger \right) \tag{C14}$$

$$+ \sum_{j=1}^{L}\sum_{k=1}^{M} \left((e^{i\varphi_j} - 1) A_{ij} C_{jk} \hat{a}_k + (e^{-i\varphi_j} - 1) B_{ij} D_{jk}^* \hat{a}_k + (e^{i\varphi_j} - 1) A_{ij} D_{jk} \hat{a}_k^\dagger + (e^{-i\varphi_j} - 1) B_{ij} C_{jk}^* \hat{a}_k^\dagger \right) \tag{C15}$$

$$= \hat{a}_i + \sum_{j=1}^{L}\sum_{k=1}^{M} \left((e^{i\varphi_j} - 1) A_{ij} C_{jk} \hat{a}_k + (e^{-i\varphi_j} - 1) B_{ij} D_{jk}^* \hat{a}_k + (e^{i\varphi_j} - 1) A_{ij} D_{jk} \hat{a}_k^\dagger + (e^{-i\varphi_j} - 1) B_{ij} C_{jk}^* \hat{a}_k^\dagger \right) \tag{C16}$$

$$= \hat{a}_i + \sum_{k=1}^{M} \left(W_{ik} \hat{a}_k + Z_{ik} \hat{a}_k^\dagger \right), \tag{C17}$$

where we defined

$$W_{ik} \equiv \sum_{j=1}^{L} \left((e^{i\varphi_j} - 1) A_{ij} C_{jk} + (e^{-i\varphi_j} - 1) B_{ij} D_{jk}^* \right) = \sum_{j=1}^{L} \left((e^{i\varphi_j} - 1) A_{ij} A_{kj}^* - (e^{-i\varphi_j} - 1) B_{ij} B_{kj}^* \right), \tag{C18}$$

$$Z_{ik} \equiv \sum_{j=1}^{L} \left((e^{i\varphi_j} - 1) A_{ij} D_{jk} + (e^{-i\varphi_j} - 1) B_{ij} C_{jk}^* \right) = \sum_{j=1}^{L} \left(-(e^{i\varphi_j} - 1) A_{ij} B_{kj} + (e^{-i\varphi_j} - 1) B_{ij} A_{kj} \right), \tag{C19}$$

and hence each of the terms has a low-rank structure. More specifically, W and Z's ranks are at most $2L$. Thus,

$$W = \sum_{j=1}^{2L} u^{(j)} v^{(j)\mathrm{T}}, \quad Z = \sum_{j=1}^{2L} w^{(j)} z^{(j)\mathrm{T}}. \tag{C20}$$

Here, note that $W + I$ and Z form a Gaussian transform. The low-rankness of the matrix Z implies that the Gaussian transform has a structure in which its squeezing parameters are nonzero only for $2L$ modes. ■

APPENDIX D: GENERALIZED GURVITS' SECOND ALGORITHM FOR LINEAR-OPTICAL CIRCUITS

In this Appendix, we present a generalized version of Gurvits' second algorithm for linear-optical circuits with an arbitrary product input, generalizing it from Fock states,

$$\langle \phi | \hat{V} | \psi \rangle = \sum_{\boldsymbol{n},\boldsymbol{m}} a_{\boldsymbol{m}} b_{\boldsymbol{n}}^* \langle \boldsymbol{n} | \hat{V} | \boldsymbol{m} \rangle, \tag{D1}$$

where $|\psi\rangle$ and $|\phi\rangle$ are arbitrary product states, which are written as

$$|\psi\rangle = \bigotimes_{i=1}^{M} \left(\sum_{m_i=0}^{n_{\max}} a_{m_i}^{(i)} |m_i\rangle \right), \quad |\phi\rangle = \bigotimes_{i=1}^{M} \left(\sum_{n_i=0}^{n_{\max}} b_{n_i}^{(i)} |n_i\rangle \right). \tag{D2}$$

Here, $\boldsymbol{n} = (n_1, \ldots, n_M)$ and $\boldsymbol{m} = (m_1, \ldots, m_M)$ and we are interested in a linear-optical circuit $\hat{V}$ whose corresponding unitary matrix V has a low-rank structure such that $V = W + I$ with a low-rank matrix W. More precisely, let us assume

that the rank of W is L, so that it can be written as

$$W_{ij} = \sum_{k=1}^{L} u_i^{(k)} v_j^{(k)}. \tag{D3}$$

To do that, let us first focus on a simpler case

$$\langle \boldsymbol{n}|\hat{V}|\boldsymbol{m}\rangle, \tag{D4}$$

which is still a more general form than that of Gurvits' algorithm because $|\boldsymbol{n}\rangle$ is not equal to $|\boldsymbol{m}\rangle$ in general. Here, because the linear-optical unitary circuit preserves the photon number, we assume that $|\boldsymbol{n}| = |\boldsymbol{m}|$; otherwise, it becomes zero. Let us introduce a notation $X_{\boldsymbol{n}}$ to represent a multiset corresponding to $\boldsymbol{n}$, which is composed of n_i times of i's. For example, for $\boldsymbol{n} = (2, 1, 0)$, $X_{\boldsymbol{n}} = \{1, 1, 2\}$.

First of all, we have

$$\langle \boldsymbol{n}|\hat{V}|\boldsymbol{m}\rangle = \frac{1}{\sqrt{\boldsymbol{n}!}}\langle 0|\left(\prod_{i\in X_{\boldsymbol{n}}}\hat{a}_i\right)\hat{V}|\boldsymbol{m}\rangle = \frac{1}{\sqrt{\boldsymbol{n}!}}\langle 0|\prod_{i\in X_{\boldsymbol{n}}}\left(\hat{a}_i + \sum_{k=1}^{M} W_{ik}\hat{a}_k\right)|\boldsymbol{m}\rangle \equiv \frac{1}{\sqrt{\boldsymbol{n}!}}\langle 0|\prod_{i\in X_{\boldsymbol{n}}}\left(\hat{a}_i + \hat{b}_i\right)|\boldsymbol{m}\rangle \tag{D5}$$

$$= \frac{1}{\sqrt{\boldsymbol{n}!}}\langle 0|\left[\sum_{X\subset X_{\boldsymbol{n}}}\prod_{x\in X}\hat{a}_x\prod_{x\in X^c}\hat{b}_x\right]|\boldsymbol{m}\rangle = \frac{1}{\sqrt{\boldsymbol{n}!}}\sum_{X\subset X_{\boldsymbol{n}}}\langle 0|\left[\prod_{x\in X}\hat{a}_x\prod_{x\in X^c}\hat{b}_x\right]|\boldsymbol{m}\rangle \tag{D6}$$

$$= \frac{1}{\sqrt{\boldsymbol{n}!}}\sum_{\boldsymbol{l}\le \boldsymbol{n},\boldsymbol{m}}\binom{\boldsymbol{n}}{\boldsymbol{l}}\frac{\sqrt{\boldsymbol{m}!}}{\sqrt{(\boldsymbol{m}-\boldsymbol{l})!}}\langle 0|\left[\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}}\sum_{j=1}^{M} W_{xj}\hat{a}_j^\dagger\right]|\boldsymbol{m}-\boldsymbol{l}\rangle \tag{D7}$$

$$= \frac{\sqrt{\boldsymbol{m}!}}{\sqrt{\boldsymbol{n}!}}\sum_{\boldsymbol{l}\le \boldsymbol{n},\boldsymbol{m}}\binom{\boldsymbol{n}}{\boldsymbol{l}}\frac{1}{(\boldsymbol{m}-\boldsymbol{l})!}\langle 0|\left[\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}}\sum_{j=1}^{M} W_{xj}\hat{a}_j^\dagger\right]\left[\prod_{x\in X_{\boldsymbol{m}-\boldsymbol{l}}}\hat{a}_x^\dagger\right]|0\rangle, \tag{D8}$$

where

$$\langle 0|\left[\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}}\sum_{j=1}^{M} W_{xj}\hat{a}_j^\dagger\right]\left[\prod_{x\in X_{\boldsymbol{m}-\boldsymbol{l}}}\hat{a}_x^\dagger\right]|0\rangle \tag{D9}$$

$$= \sum_{\sigma\in\mathcal{S}_{|X_{\boldsymbol{n}-\boldsymbol{l}}|}:X_{\boldsymbol{n}-\boldsymbol{l}}\to X_{\boldsymbol{m}-\boldsymbol{l}}}\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}} W_{x,\sigma(x)} \tag{D10}$$

$$= \sum_{\sigma\in\mathcal{S}_{|X_{\boldsymbol{n}-\boldsymbol{l}}|}:X_{\boldsymbol{n}-\boldsymbol{l}}\to X_{\boldsymbol{m}-\boldsymbol{l}}}\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}}\sum_{k=1}^{L} u_x^{(k)} v_{\sigma(x)}^{(k)} \tag{D11}$$

$$= \sum_{\sigma\in\mathcal{S}_{|X_{\boldsymbol{n}-\boldsymbol{l}}|}:X_{\boldsymbol{n}-\boldsymbol{l}}\to X_{\boldsymbol{m}-\boldsymbol{l}}}\sum_{k_1,\ldots,k_{|X_{\boldsymbol{n}-\boldsymbol{l}}|}=1}^{L}\prod_{i=1}^{|X_{\boldsymbol{n}-\boldsymbol{l}}|} u_{X_{\boldsymbol{n}-\boldsymbol{l}}(i)}^{(k_i)} v_{X_{\boldsymbol{m}-\boldsymbol{l}}(i)}^{(k_{\sigma(i)})} \tag{D12}$$

$$= \sum_{s_u(X_{\boldsymbol{n}-\boldsymbol{l}})}\sum_{s_v(X_{\boldsymbol{m}-\boldsymbol{l}})}\delta([s_u(X_{\boldsymbol{n}-\boldsymbol{l}})] = [s_v(X_{\boldsymbol{m}-\boldsymbol{l}})])\prod_{i=1}^{|X_{\boldsymbol{n}-\boldsymbol{l}}|} u_{X_{\boldsymbol{n}-\boldsymbol{l}}(i)}^{(s_u(i))} v_{X_{\boldsymbol{m}-\boldsymbol{l}}(i)}^{(s_v(i))}\prod_{i=1}^{L} k_i! \tag{D13}$$

$$= \sum_{s_u(X_{\boldsymbol{n}-\boldsymbol{l}})}\sum_{s_v(X_{\boldsymbol{m}-\boldsymbol{l}})}\delta([s_u(X_{\boldsymbol{n}-\boldsymbol{l}})] = [s_v(X_{\boldsymbol{m}-\boldsymbol{l}})])\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}} u_x^{(s_u(x))}\prod_{x\in X_{\boldsymbol{m}-\boldsymbol{l}}} v_x^{(s_v(x))}\prod_{i=1}^{L} k_i!, \tag{D14}$$

where the summation over $s(X)$ is interpreted as the summation of a tuple from $[L]^{|X|}$ and $[s(X)]$ denotes the multiset of the elements of $s(X)$, and k_i's denote the multiplicity of i in the multiset $[s_u(X)]$, which arises because of the redundancy.

At this moment, let us ignore the delta function and the multiplicity factor $\prod_i k_i!$, which will be taken care of later. The full expression is then rewritten as

$$\frac{\sqrt{\boldsymbol{m}!}}{\sqrt{\boldsymbol{n}!}} \sum_{\boldsymbol{l}\le \boldsymbol{n},\boldsymbol{m}} \binom{\boldsymbol{n}}{\boldsymbol{l}} \frac{1}{(\boldsymbol{m}-\boldsymbol{l})!} \sum_{s_u(X_{\boldsymbol{n}-\boldsymbol{l}})} \sum_{s_v(X_{\boldsymbol{m}-\boldsymbol{l}})} \prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}} u_x^{(s_u(x))} \prod_{x\in X_{\boldsymbol{m}-\boldsymbol{l}}} v_x^{(s_v(x))} \tag{D15}$$

$$= \frac{\sqrt{\boldsymbol{m}!}}{\sqrt{\boldsymbol{n}!}} \sum_{\boldsymbol{l}\le \boldsymbol{n},\boldsymbol{m}} \binom{\boldsymbol{n}}{\boldsymbol{l}} \frac{1}{(\boldsymbol{m}-\boldsymbol{l})!} \left(\prod_{x\in X_{\boldsymbol{n}-\boldsymbol{l}}} \sum_{s=1}^{L} u_x^{(s)}\right) \left(\prod_{x\in X_{\boldsymbol{m}-\boldsymbol{l}}} \sum_{s=1}^{L} v_x^{(s)}\right) \tag{D16}$$

$$= \frac{\sqrt{\boldsymbol{m}!}}{\sqrt{\boldsymbol{n}!}} \prod_{i=1}^{M} \sum_{l\le \min(n_i,m_i)} \binom{n_i}{l} \frac{1}{(m_i-l)!} \left(\sum_{s=1}^{L} u_i^{(s)}\right)^{n_i-l} \left(\sum_{s=1}^{L} v_i^{(s)}\right)^{m_i-l}. \tag{D17}$$

Now, to take care of the delta function, we introduce the indeterminates $\{a_u^{(s)}\}_{s=1}^L$ and $\{a_v^{(s)}\}_{s=1}^L$ and write the expression as

$$\prod_{i=1}^{M} \frac{\sqrt{m_i!}}{\sqrt{n_i!}} \sum_{l\le \min(n_i,m_i)} \binom{n_i}{l} \frac{1}{(m_i-l)!} \left(\sum_{s=1}^{L} a_u^{(s)} u_x^{(s)}\right)^{n_i-l} \left(\sum_{s=1}^{L} a_v^{(s)} v_i^{(s)}\right)^{m_i-l} \tag{D18}$$

and keep the coefficients of the monomials in which $a_u^{(s)}$ and $a_v^{(s)}$'s degrees match each other [e.g., $(a_u^{(1)})^2(a_u^{(2)})^3(a_v^{(1)})^2(a_v^{(2)})^3$]. Finally, to take care of the multiplicity factor $\prod_i k_i$, we then multiply the factorial of degrees of $a_u^{(s)}$ (or equivalently $a_v^{(s)}$).

Now, for arbitrary product input states $|\psi\rangle$ and $|\phi\rangle$, we have

$$\langle\phi|\hat{V}|\psi\rangle = \sum_{\boldsymbol{n},\boldsymbol{m}} a_{\boldsymbol{m}} b_{\boldsymbol{n}}^* \langle \boldsymbol{n}|\hat{V}|\boldsymbol{m}\rangle \tag{D19}$$

$$\to \sum_{\boldsymbol{n},\boldsymbol{m}} a_{\boldsymbol{m}} b_{\boldsymbol{n}}^* \prod_{i=1}^{M} \frac{\sqrt{m_i!}}{\sqrt{n_i!}} \sum_{l\le \min(n_i,m_i)} \binom{n_i}{l} \frac{1}{(m_i-l)!} \left(\sum_{s=1}^{L} a_u^{(s)} u_i^{(s)}\right)^{n_i-l} \left(\sum_{s=1}^{L} a_v^{(s)} v_i^{(s)}\right)^{m_i-l} \tag{D20}$$

$$\to \prod_{i=1}^{M} \sum_{n_i,m_i=0}^{n_{\max}} a_{m_i}^{(i)} b_{n_i}^{(i)*} \frac{\sqrt{m_i!}}{\sqrt{n_i!}} \sum_{l\le \min(n_i,m_i)} \binom{n_i}{l} \frac{1}{(m_i-l)!} \left(\sum_{s=1}^{L} a_u^{(s)} u_i^{(s)}\right)^{n_i-l} \left(\sum_{s=1}^{L} a_v^{(s)} v_i^{(s)}\right)^{m_i-l}. \tag{D21}$$

Let us now analyze the complexity. The number of indeterminates we introduce is $2L$, and each indeterminate may have degree at most $Mn_{\max}$. Thus, the number of coefficients we must keep track of is $(Mn_{\max}+1)^{2L}$. Since there are M product, the total number of operations is $O(M(Mn_{\max}+1)^{2L})$.

APPENDIX E: ANTINORMAL ORDERING

The goal of this Appendix is to find the antinormal ordering of the following term:

$$\prod_{i=1}^{N} (\hat{A}_i + \hat{B}_i^\dagger), \tag{E1}$$

where the commutation relations of the operators are given as $[\hat{A}_i, \hat{A}_j] = 0$, $[\hat{B}_i^\dagger, \hat{B}_j^\dagger] = 0$, and $[\hat{A}_i, \hat{B}_j^\dagger] = C_{ij}$. This Appendix is based on Ref. [92]. To find its antinormal ordering, i.e., express the above by operators in which $\hat{A}_i$'s always appear on the left and $\hat{B}_i^\dagger$'s always appear on the right, we consider each term after expanding it, such as $\hat{A}_1\hat{B}_2^\dagger\hat{A}_3\hat{A}_4\hat{B}_5^\dagger\hat{A}_6$. Now, we move all the creation operators $\hat{B}_i^\dagger$'s to the right by using the commutation relation. First, we denote Y as the indices of the annihilation operators $\hat{A}_i$ and Y^c as those of the creation operators $\hat{B}_i^\dagger$. For example, for $\hat{A}_1\hat{B}_2^\dagger\hat{A}_3\hat{A}_4\hat{B}_5^\dagger\hat{A}_6$, $Y = \{1,3,4,6\}$ and $Y^c = \{2,5\}$. We then move $\hat{B}_i^\dagger$ for $i \in Y^c$ one by one using the commutation relation. First of all, note that when we move $\hat{B}_j^\dagger$ to the right, it only interacts with $\hat{A}_i$ with $i > j$.

Let us start with the largest $j \in Y^c$, say 5 for the above example. Using $\hat{B}_5^\dagger \hat{A}_6 = [\hat{B}_5^\dagger, \hat{A}_6] + \hat{A}_6 \hat{B}_5^\dagger = -C_{65} + \hat{A}_6 \hat{B}_5^\dagger$, we now have two terms,

$$\hat{A}_1 \hat{B}_2^\dagger \hat{A}_3 \hat{A}_4 \hat{B}_5^\dagger \hat{A}_6 = -C_{65} \hat{A}_1 \hat{B}_2^\dagger \hat{A}_3 \hat{A}_4 + \hat{A}_1 \hat{B}_2^\dagger \hat{A}_3 \hat{A}_4 \hat{A}_6 \hat{B}_5^\dagger. \tag{E2}$$

Thus, when we use the commutation relation between $\hat{A}_i$ and $\hat{B}_j^\dagger$, the associated term is split into two terms: one that changes $\hat{A}_i \hat{B}_j^\dagger$ to $-C_{ij}$ and the other that simply changes $\hat{A}_i \hat{B}_j^\dagger$ to $\hat{B}_j^\dagger \hat{A}_i$. Using this property, one can easily see that for a given Y and Y^c, its antinormal ordering is simply written as

$$\sum_{\mathcal{M} \in \mathrm{Match}(F_N; Y \to Y^c)} (-1)^{|\mathcal{M}|} \prod_{(i,j) \in \mathcal{M}} C_{ij} \prod_{i \in Y \setminus A(\mathcal{M})} \hat{A}_i \prod_{j \in Y^c \setminus B(\mathcal{M})} \hat{B}_j^\dagger, \tag{E3}$$

where the Match is the possible matches in F_N, which is the Ferrers graph whose edge set is $E = \{(i,j) | 1 \le j < i \le N\}$ [93], and $A(\mathcal{M})$ and $B(\mathcal{M})$ represent the elements in Y and Y^c that are involved in the matching $\mathcal{M}$, respectively.

Since we need to take into account all possible pairs Y and Y^c, we have

$$\prod_{i=1}^{N} (\hat{A}_i + \hat{B}_i^\dagger) = \sum_{Y \subset [N]} \sum_{\mathcal{M} \in \mathrm{Match}(F_N; Y \to Y^c)} (-1)^{|\mathcal{M}|} \prod_{(i,j) \in \mathcal{M}} C_{ij} \prod_{i \in Y \setminus A(\mathcal{M})} \hat{A}_i \prod_{j \in Y^c \setminus B(\mathcal{M})} \hat{B}_j^\dagger. \tag{E4}$$

APPENDIX F: GENERALIZED GURVITS' SECOND ALGORITHM FOR GAUSSIAN CIRCUITS

In this Appendix, we provide a classical algorithm that computes the following using a low-rank property of the relevant matrices:

$$\langle \phi | \hat{G} | \psi \rangle = \sum_{\boldsymbol{n}, \boldsymbol{m}} a_{\boldsymbol{m}} b_{\boldsymbol{n}}^* \langle \boldsymbol{n} | \hat{G} | \boldsymbol{m} \rangle, \tag{F1}$$

where $|\psi\rangle$ and $|\phi\rangle$ are arbitrary product states, which are written due to the product structure as

$$|\psi\rangle = \bigotimes_{i=1}^{M} \left(\sum_{m_i=0}^{n_{\max}} a_{m_i}^{(i)} |m_i\rangle \right), \quad |\phi\rangle = \bigotimes_{i=1}^{M} \left(\sum_{n_i=0}^{n_{\max}} b_{n_i}^{(i)} |n_i\rangle \right). \tag{F2}$$

Also, $\hat{G}$ is a Gaussian unitary circuit, and in particular, we are interested in a Gaussian unitary circuit generated by $\hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0$, where $\hat{G}_0$ is an arbitrary Gaussian unitary circuit and $\hat{P}(\boldsymbol{\varphi})$ is a phase shifter with phases $\boldsymbol{\varphi}$ whose nonzero elements are a few, i.e., $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_L, 0, \ldots, 0)$. Hence, we prove Lemma 3:

Lemma F1 (Lemma 3 restated). Consider an M-mode bosonic system. Let $\hat{G}_0$ be a Gaussian unitary circuit and $\hat{P}(\boldsymbol{\varphi})$ be the phase shifter with phase $\boldsymbol{\varphi}$ and $|\psi\rangle$ and $|\phi\rangle$ be product states. If the number of nonzero elements in $\boldsymbol{\varphi}$ is $L = O(1)$,

$$\langle \phi | \hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0 | \psi \rangle \tag{F3}$$

can be exactly computed in polynomial time in M.

Proof. Before we derive the full expression and the classical algorithm, let us first focus on $\langle \boldsymbol{n} | \hat{G} | \boldsymbol{m} \rangle$ and then generalize, where $\hat{G} \equiv \hat{G}_0^\dagger \hat{P}(\boldsymbol{\varphi}) \hat{G}_0$. Let $X_{\boldsymbol{n}}$ be the multiset representation of $\boldsymbol{n}$, which is composed of n_i times of i. For example, when $\boldsymbol{n} = (2, 1, 4)$, $X_{\boldsymbol{n}} = \{1, 1, 2, 3, 3, 3, 3\}$. $X_{\boldsymbol{m}}$ is similar. Let $\hat{G} = \hat{U} \hat{S}(\boldsymbol{r}) \hat{V}$ be the Bloch-Messiah decomposition of $\hat{G}$,

following Appendix C. Then,

$$\langle \boldsymbol{n}|\hat{G}|\boldsymbol{m}\rangle = \frac{1}{\sqrt{\boldsymbol{n}!}}\langle 0|\hat{G}\hat{G}^\dagger\left(\prod_{i\in X_{\boldsymbol{n}}}\hat{a}_i\right)\hat{G}|\boldsymbol{m}\rangle = \frac{1}{\sqrt{\boldsymbol{n}!}}\langle 0|\hat{G}\left[\prod_{i\in X_{\boldsymbol{n}}}\left(\hat{a}_i+\sum_{k=1}^{M}\left(W_{ik}\hat{a}_k+Z_{ik}\hat{a}_k^\dagger\right)\right)\right]|\boldsymbol{m}\rangle \tag{F4}$$

$$= \frac{1}{\sqrt{\boldsymbol{n}!}}\langle \boldsymbol{r}|\hat{V}\left[\prod_{i\in X_{\boldsymbol{n}}}\left(\hat{a}_i+\sum_{k=1}^{M}\left(W_{ik}\hat{a}_k+Z_{ik}\hat{a}_k^\dagger\right)\right)\right]|\boldsymbol{m}\rangle = \frac{1}{\sqrt{\boldsymbol{n}!}}\langle \boldsymbol{r}|\hat{V}\left[\prod_{i\in X_{\boldsymbol{n}}}\left(\hat{A}_i+\hat{B}_i^\dagger\right)\right]|\boldsymbol{m}\rangle, \tag{F5}$$

where we defined

$$\hat{A}_i = \hat{a}_i+\sum_{k=1}^{M}W_{ik}\hat{a}_k,\quad \hat{B}_i^\dagger=\sum_{k=1}^{M}Z_{ik}\hat{a}_k^\dagger,\quad C_{ij}=[\hat{A}_i,\hat{B}_j^\dagger]=Z_{ji}+\sum_{k=1}^{M}W_{ik}Z_{jk}. \tag{F6}$$

Note that, as shown in Appendix C, W, Z, and C's ranks are at most $2L$ and that only at most $2L$ r_i's are nonzero. Without loss of generality, we set $r_i = 0$ for $i > 2L$.

By using the antinormal ordering [92] (see Appendix E), let us expand it and move all the creation operators to the right,

$$\frac{1}{\sqrt{\boldsymbol{n}!\boldsymbol{m}!}}\sum_{Y\subset X_{\boldsymbol{n}}}\sum_{\mathcal{M}\in\mathrm{Match}(F_{X_{\boldsymbol{n}}};Y\to Y^c)}(-1)^{|\mathcal{M}|}\left(\prod_{(i,j)\in\mathcal{M}}C_{ij}\right)\langle \boldsymbol{r}|\hat{V}\left(\prod_{i\in Y\setminus A(\mathcal{M})}\hat{A}_i\right)\left(\prod_{i\in Y^c\setminus B(\mathcal{M})}\hat{B}_i^\dagger\right)\prod_{i\in X_{\boldsymbol{m}}}\hat{a}_i^\dagger|0\rangle, \tag{F7}$$

where Y and $Y^c\equiv X_{\boldsymbol{n}}\setminus Y$ are multisets that are subsets of $X_{\boldsymbol{n}}$ and the Match is the possible matches in $F_{X_{\boldsymbol{n}}}$, which is the Ferrers graph whose edge set is $E=\{(i,j)\in X_{\boldsymbol{n}}\times X_{\boldsymbol{n}}|j<i\}$. Also, $A(\mathcal{M})$ and $B(\mathcal{M})$ are the elements that are involved in M from Y and Y^c, respectively. That means that $Y\setminus A(\mathcal{M})$ and $Y^c\setminus B(\mathcal{M})$ are vertices that are not involved in the matching $\mathcal{M}$. Here,

$$\langle \boldsymbol{r}|\hat{V}\left(\prod_{i\in Y\setminus A(\mathcal{M})}\hat{A}_i\right)\left(\prod_{i\in Y^c\setminus B(\mathcal{M})}\hat{B}_i^\dagger\right)\prod_{i\in X_{\boldsymbol{m}}}\hat{a}_i^\dagger|0\rangle \tag{F8}$$

$$= \langle \boldsymbol{r}|\hat{V}\sum_{X\subset Y\setminus A(\mathcal{M})}\prod_{i\in X}\hat{a}_i\left(\prod_{i\in Y\setminus A(\mathcal{M})\setminus X}\sum_{k=1}^{M}W_{ik}\hat{a}_k\right)\left(\prod_{i\in Y^c\setminus B(\mathcal{M})}\sum_{k=1}^{M}Z_{jk}\hat{a}_k^\dagger\right)\left(\prod_{i\in X_{\boldsymbol{m}}}\hat{a}_i^\dagger\right)|0\rangle. \tag{F9}$$

Thus, let us first simplify the following:

$$\langle \boldsymbol{r}|\hat{V}\prod_{i\in A}\hat{a}_i\prod_{i\in B}\hat{b}_i\prod_{i\in C}\hat{c}_i^\dagger\prod_{i\in D}\hat{a}_i^\dagger|0\rangle, \tag{F10}$$

where $A\iff X, B\iff Y\setminus A(\mathcal{M})\setminus X, C\iff Y^c\setminus B(\mathcal{M}), D\iff X_{\boldsymbol{m}}$. Here,

$$\prod_{i\in A}\hat{a}_i\prod_{i\in B}\hat{b}_i\prod_{i\in C}\hat{c}_i^\dagger\prod_{i\in D}\hat{a}_i^\dagger|0\rangle \tag{F11}$$

$$= \prod_{i\in A}\hat{a}_i\prod_{i\in B}\sum_{j=1}^{M}W_{ij}\hat{a}_j\prod_{i\in C}\sum_{j=1}^{M}Z_{ij}\hat{a}_j^\dagger\prod_{i\in D}\hat{a}_i^\dagger|0\rangle \tag{F12}$$

$$= \sum_{j_{B(1)},\ldots,j_{B(|B|)}=1}^{M}\sum_{j'_{C(1)},\ldots,j'_{C(|C|)}=1}^{M}\prod_{i\in B}W_{ij_i}\prod_{i\in C}Z_{ij'_i}\prod_{i\in A}\hat{a}_i\prod_{i\in B}\hat{a}_{j_i}\prod_{i\in C}\hat{a}_{j'_i}^\dagger\prod_{i\in D}\hat{a}_i^\dagger|0\rangle. \tag{F13}$$

To simplify this expression, we first need to match all A and B to C and D first, and then the remaining elements in C and D will be matched with the squeezed state $\langle \boldsymbol{r}|\hat{V}$. Here, $a\subset A$ matches with $a=d_a\subset D$ and $b\subset B$ with $d_b\subset D\setminus a$, and

$A \setminus a$ with $c_a \subset C$ and $B \setminus b$ with c_b. Thus,

$$\prod_{i\in A}\hat{a}_i \prod_{i\in B}\hat{a}_{j_i} \prod_{i\in C}\hat{a}^\dagger_{j'_i} \prod_{i\in D}\hat{a}^\dagger_i|0\rangle \tag{F14}$$

$$= \sum_{a\subset A\cap D}\sum_{b\subset B}\sum_{c_a,c_b\subset C}\sum_{d_b\subset D\setminus d_a}\sum_{\sigma\in\mathcal{S}_{|a|}:a\to d_a}\prod_{i\in a}\delta_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\prod_{i\in a^c}\delta_{i,j'_{\sigma(i)}}\sum_{\sigma\in\mathcal{S}_{|b|}:b\to d_b}\prod_{i\in b}\delta_{j_i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\prod_{i\in B\setminus b}\delta_{j_i,j'_{\sigma(i)}}$$

$$\times \prod_{i\in C\setminus(c_a\cup c_b)}\hat{a}^\dagger_{j'_i}\prod_{i\in D\setminus(d_a\cup d_b)}\hat{a}^\dagger_i|0\rangle. \tag{F15}$$

Hence, we have

$$\langle \boldsymbol{r}|\hat{V}\sum_{j_{B(1)},\dots,j_{B(|B|)}=1}^{M}\sum_{j'_{C(1)},\dots,j'_{C(|C|)}=1}^{M}\prod_{i\in B}W_{ij_i}\prod_{i\in C}Z_{ij'_i}\prod_{i\in A}\hat{a}_i\prod_{i\in B}\hat{a}_{j_i}\prod_{i\in C}\hat{a}^\dagger_{j'_i}\prod_{i\in D}\hat{a}^\dagger_i|0\rangle \tag{F16}$$

$$= \sum_{a\subset A\cap D}\sum_{b\subset B}\sum_{c_a,c_b\subset C}\sum_{d_b\subset D\setminus d_a}\sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\prod_{i\in a^c}Z_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|b|}:b\to d_b}\prod_{i\in b}W_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\prod_{i\in B\setminus b}(WZ^{\mathrm{T}})_{i,\sigma(i)}$$

$$\times \langle \boldsymbol{r}|\hat{V}\prod_{i\in C\setminus(c_a\cup c_b)}\sum_{j'_i=1}^{M}Z_{ij'_i}\hat{a}^\dagger_{j'_i}\prod_{i\in D\setminus(d_a\cup d_b)}\hat{a}^\dagger_i|0\rangle \tag{F17}$$

$$= \sum_{a\subset A\cap D}\sum_{b\subset B}\sum_{c_a,c_b\subset C}\sum_{d_b\subset D\setminus d_a}\sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\prod_{i\in a^c}Z_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|b|}:b\to d_b}\prod_{i\in b}W_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\prod_{i\in B\setminus b}(WZ^{\mathrm{T}})_{i,\sigma(i)}$$

$$\times \langle \boldsymbol{r}|\prod_{i\in C\setminus(c_a\cup c_b)}\sum_{j'_i,k=1}^{M}Z_{ij'_i}V_{k,j'_i}\hat{a}^\dagger_k\prod_{i\in D\setminus(d_a\cup d_b)}\sum_{k=1}^{M}V_{k,i}\hat{a}^\dagger_k|0\rangle \tag{F18}$$

$$= \sum_{a\subset A\cap D}\sum_{b\subset B}\sum_{c_a,c_b\subset C}\sum_{d_b\subset D\setminus d_a}\sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\prod_{i\in a^c}Z_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|b|}:b\to d_b}\prod_{i\in b}W_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\prod_{i\in B\setminus b}(WZ^{\mathrm{T}})_{i,\sigma(i)}$$

$$\times \langle \boldsymbol{r}|\prod_{i\in C\setminus(c_a\cup c_b)}\sum_{k=1}^{M}(ZV^{\mathrm{T}})_{i,k}\hat{a}^\dagger_k\prod_{i\in D\setminus(d_a\cup d_b)}\sum_{k=1}^{M}(V^{\mathrm{T}})_{i,k}\hat{a}^\dagger_k|0\rangle, \tag{F19}$$

where $c_a \cap c_b = \emptyset$ and $d_a \cap d_b = \emptyset$. Here, using the low-rank structure of W and Z, we simplify the expression further as

$$\sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\prod_{i\in a^c}Z_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|b|}:b\to d_b}\prod_{i\in b}W_{i,\sigma(i)}\sum_{\sigma\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\prod_{i\in B\setminus b}(WZ^{\mathrm{T}})_{i,\sigma(i)} \tag{F20}$$

$$= \sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\prod_{i\in a^c}\sum_{l=1}^{2L}u_i^{(l)}v_{\sigma(i)}^{(l)}\sum_{\rho\in\mathcal{S}_{|b|}:b\to d_b}\prod_{i\in b}\sum_{k=1}^{2L}w_i^{(k)}x_{\rho(i)}^{(k)}\sum_{\omega\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\prod_{i\in B\setminus b}\sum_{m=1}^{2L}y_i^{(m)}z_{\omega(i)}^{(m)} \tag{F21}$$

$$= \sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\sum_{l_{a^c(1)},\dots,l_{a^c(|a^c|)}=1}^{2L}\prod_{i\in a^c}u_i^{(l_i)}v_{\sigma(i)}^{(l_i)}\sum_{\rho\in\mathcal{S}_{|b|}:b\to d_b}\sum_{k_{b(1)},\dots,k_{b(|b|)}=1}^{2L}\prod_{i\in b}w_i^{(k_i)}x_{\rho(i)}^{(k_i)}$$

$$\times \sum_{\omega\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\sum_{m_{(B\setminus b)(1)},\dots,m_{(B\setminus b)(|B\setminus b|)}=1}^{2L}\prod_{i\in B\setminus b}y_i^{(m_i)}z_{\omega(i)}^{(m_i)} \tag{F22}$$

$$= \sum_{\sigma\in\mathcal{S}_{|a^c|}:a^c\to c_a}\sum_{\rho\in\mathcal{S}_{|b|}:b\to d_b}\sum_{\omega\in\mathcal{S}_{|B\setminus b|}:B\setminus b\to c_b}\sum_{l_1,\dots,l_{|a^c|}=1}^{2L}\sum_{k_1,\dots,k_{|b|}=1}^{2L}\sum_{m_1,\dots,m_{|B\setminus b|}=1}^{2L}\prod_{i\in[a^c]}u_{c_a(i)}^{(l_i)}v_{a^c(i)}^{(l_{\sigma(i)})}\prod_{i\in[b]}w_{b(i)}^{(k_{\rho(i)})}x_{d_b(i)}^{(k_i)}$$

$$\times \prod_{i\in[B\setminus b]}y_{c_b(i)}^{(m_i)}z_{(B\setminus b)(i)}^{(m_{\omega(i)})} \tag{F23}$$

$$= \sum_{s_u(c_a),s_v(a^c)} \sum_{s_w(b),s_x(d_b)} \sum_{s_z(B\setminus b),s_y(c_b)} \delta([s_u(c_a)] = [s_v(a^c)])\delta([s_w(b)] = [s_x(d_b)])\delta([s_z(B\setminus b)] = [s_y(c_b)])$$

$$\times \prod_{i\in[a^c]} u_{c_a(i)}^{(l_{s_u(i)})} v_{a^c(i)}^{(s_v(i))} \prod_{i\in[b]} w_{b(i)}^{(s_w(i))} x_{d_b(i)}^{(k_{s_x(i)})} \prod_{i\in[B\setminus b]} y_{c_b(i)}^{(s_z(i))} z_{(B\setminus b)(i)}^{(s_y(i))} \prod_{i=1}^{L} (k_i!k_i'!k_i''!), \tag{F24}$$

where the summation over $s(X)$ is interpreted as the summation of a tuple from $[L]^{|X|}$ and k_i, k_i', k_i'' represent the multiplicity of u, w, y, respectively. Thus, ignoring the delta functions and the multiplicity factors, which will be taken care of later, we can rewrite the expression as

$$\sum_{a\subset A\cap D} \sum_{b\subset B} \sum_{c_a,c_b\subset C} \sum_{d_b\subset D\setminus a} \sum_{s_u(c_a),s_v(a^c)} \sum_{s_w(b),s_x(d_b)} \sum_{s_z(B\setminus b),s_y(c_b)} \prod_{i\in[a^c]} u_{c_a(i)}^{(l_{s_u(i)})} v_{a^c(i)}^{(s_v(i))} \prod_{i\in[b]} w_{b(i)}^{(s_w(i))} x_{d_b(i)}^{(k_{s_x(i)})} \prod_{i\in[B\setminus b]} y_{c_b(i)}^{(s_z(i))} z_{(B\setminus b)(i)}^{(s_y(i))} \tag{F25}$$

$$= \sum_{a\subset A\cap D} \sum_{b\subset B} \sum_{c_a,c_b\subset C} \sum_{d_b\subset D\setminus a} \prod_{i\in c_a} \sum_{s=1}^{2L} u_i^{(s)} a_u^{(s)} \prod_{i\in a^c} \sum_{s=1}^{2L} v_i^{(s)} a_v^{(s)} \prod_{i\in b} \sum_{s=1}^{2L} w_i^{(s)} a_w^{(s)} \prod_{i\in d_b} \sum_{s=1}^{2L} x_i^{(s)} a_x^{(s)} \prod_{i\in c_b} \sum_{s=1}^{2L} y_i^{(s)} a_y^{(s)} \prod_{i\in B\setminus b} \sum_{s=1}^{2L} z_i^{(s)} a_z^{(s)} \tag{F26}$$

$$= \sum_{a\subset A\cap D} \sum_{b\subset B} \sum_{c_a,c_b\subset C} \sum_{d_b\subset D\setminus a} \prod_{i\in c_a} f_1(i) \prod_{i\in a^c} f_2(i) \prod_{i\in b} f_3(i) \prod_{i\in d_b} f_4(i) \prod_{i\in c_b} f_5(i) \prod_{i\in B\setminus b} f_6(i), \tag{F27}$$

where we introduced the indeterminates $a_u^{(s)}, a_v^{(s)}, a_w^{(s)}, a_x^{(s)}, a_y^{(s)}, a_z^{(s)}$ to take care of the delta functions and defined

$$f_1(i) = \sum_{s=1}^{2L} u_i^{(s)} a_u^{(s)}, \quad f_2(i) = \sum_{s=1}^{2L} v_i^{(s)} a_v^{(s)}, \quad f_3(i) = \sum_{s=1}^{2L} w_i^{(s)} a_w^{(s)}, \quad f_4(i) = \sum_{s=1}^{2L} x_i^{(s)} a_x^{(s)}, \quad f_5(i) = \sum_{s=1}^{2L} y_i^{(s)} a_y^{(s)},$$

$$f_6(i) = \sum_{s=1}^{2L} z_i^{(s)} a_z^{(s)}. \tag{F28}$$

In addition, to take care of the remaining creation operators, we replace them with indeterminates x_k's as in Sec. VI B 2,

$$\sum_{a\subset A\cap D} \sum_{b\subset B} \sum_{c_a,c_b\subset C} \sum_{d_b\subset D\setminus a} \prod_{i\in c_a} f_1(i) \prod_{i\in a^c} f_2(i) \prod_{i\in b} f_3(i) \prod_{i\in d_b} f_4(i) \prod_{i\in c_b} f_5(i) \prod_{i\in B\setminus b} f_6(i)$$

$$\times \langle \boldsymbol{r}| \prod_{i\in C\setminus(c_a\cup c_b)} \sum_{k=1}^{M} (ZV^{\mathrm{T}})_{i,k} \hat{a}_k^\dagger \prod_{i\in D\setminus(d_a\cup d_b)} \sum_{k=1}^{M} (V^{\mathrm{T}})_{i,k} \hat{a}_k^\dagger |0\rangle \tag{F29}$$

$$\to \sum_{a\subset A\cap D} \sum_{b\subset B} \sum_{c_a,c_b\subset C} \sum_{d_b\subset D\setminus a} \prod_{i\in c_a} f_1(i) \prod_{i\in a^c} f_2(i) \prod_{i\in b} f_3(i) \prod_{i\in d_b} f_4(i) \prod_{i\in c_b} f_5(i) \prod_{i\in B\setminus b} f_6(i)$$

$$\times \prod_{i\in C\setminus(c_a\cup c_b)} \sum_{k=1}^{2L} (ZV^{\mathrm{T}})_{i,k} x_k \prod_{i\in D\setminus(d_a\cup d_b)} \sum_{k=1}^{2L} (V^{\mathrm{T}})_{i,k} x_k \tag{F30}$$

$$\to \sum_{a\subset A\cap D} \sum_{b\subset B} \sum_{c_a,c_b\subset C} \sum_{d_b\subset D\setminus a} \prod_{i\in c_a} f_1(i) \prod_{i\in a^c} f_2(i) \prod_{i\in b} f_3(i) \prod_{i\in d_b} f_4(i) \prod_{i\in c_b} f_5(i) \prod_{i\in B\setminus b} f_6(i) \prod_{i\in C\setminus(c_a\cup c_b)} g_1(i) \prod_{i\in D\setminus(d_a\cup d_b)} g_2(i), \tag{F31}$$

where we defined

$$g_1(i) = \sum_{k=1}^{2L} (ZV^{\mathrm{T}})_{i,k} x_k, \quad g_2(i) = \sum_{k=1}^{2L} (V^{\mathrm{T}})_{i,k} x_k. \tag{F32}$$

The squeezed states' coefficients will be taken care of later by using the indeterminates x_k's.

Now we simplify the expression. Using

$$\sum_{b\subset B}\left(\prod_{i\in b}f_3(i)\prod_{i\in B\setminus b}f_6(i)\right)=\prod_{i\in B}(f_3(i)+f_6(i)), \tag{F33}$$

$$\sum_{c_a,c_b\subset C}\left(\prod_{i\in c_a}f_1(i)\prod_{i\in c_b}f_5(i)\prod_{i\in C\setminus(c_a\cup c_b)}g_1(i)\right)=\prod_{i\in C}(f_1(i)+f_5(i)+g_1(i)), \tag{F34}$$

$$\sum_{a\subset A\cap D}\sum_{d_b\subset D\setminus a}\left(\prod_{i\in A\setminus a}f_2(i)\prod_{i\in d_b}f_4(i)\prod_{i\in(D\setminus a)\setminus d_b}g_2(i)\right) \tag{F35}$$

$$=\prod_{i\in A\setminus D}f_2(i)\sum_{a\subset A\cap D}\prod_{i\in(A\cap D)\setminus a}f_2(i)\sum_{d_b\subset D\setminus a}\left(\prod_{i\in d_b}f_4(i)\prod_{i\in(D\setminus a)\setminus d_b}g_2(i)\right) \tag{F36}$$

$$=\prod_{i\in A\setminus D}f_2(i)\sum_{a\subset A\cap D}\prod_{i\in(A\cap D)\setminus a}f_2(i)\prod_{i\in D\setminus a}(f_4(i)+g_2(i)) \tag{F37}$$

$$=\prod_{i\in A\setminus D}f_2(i)\prod_{i\in D\setminus A}(f_4(i)+g_2(i))\sum_{a\subset A\cap D}\prod_{i\in(A\cap D)\setminus a}f_2(i)\prod_{i\in(A\cap D)\setminus a}(f_4(i)+g_2(i)) \tag{F38}$$

$$=\prod_{i\in A\setminus D}f_2(i)\prod_{i\in D\setminus A}(f_4(i)+g_2(i))\prod_{i\in A\cap D}[1+f_2(i)(f_4(i)+g_2(i))], \tag{F39}$$

the expression is simplified as

$$\prod_{i\in A\setminus D}f_2(i)\prod_{i\in D\setminus A}(f_4(i)+g_2(i))\prod_{i\in A\cap D}[1+f_2(i)(f_4(i)+g_2(i))]\prod_{i\in B}(f_3(i)+f_6(i))\prod_{i\in C}(f_1(i)+f_5(i)+g_1(i)). \tag{F40}$$

And recalling

$$A\iff X,\quad B\iff Y\setminus A(\mathcal{M})\setminus X,\quad C\iff Y^c\setminus B(\mathcal{M}),\quad D\iff X_{\boldsymbol{m}}, \tag{F41}$$

the full expression becomes

$$\prod_{i\in X\setminus X_{\boldsymbol{m}}}F_i\prod_{i\in X_{\boldsymbol{m}}\setminus X}E_i\prod_{i\in X\cap X_{\boldsymbol{m}}}D_i\prod_{i\in Y\setminus A(\mathcal{M})\setminus X}A_i\prod_{i\in Y^c\setminus B(\mathcal{M})}B_i, \tag{F42}$$

where we defined

$$A_i=f_3(i)+f_6(i),\quad B_i=f_1(i)+f_5(i)+g_1(i),\quad D_i=1+f_2(i)(f_4(i)+g_2(i)),\quad E_i=f_4(i)+g_2(i),\quad F_i=f_2(i). \tag{F43}$$

Thus, we can simplify the expression as a loop hafnian,

$$\sum_{Y\subset X_{\boldsymbol{n}}}\sum_{\mathcal{M}\in\mathrm{Match}(F_{X_{\boldsymbol{n}}};Y\to Y^c)}\sum_{X\subset Y\setminus A(\mathcal{M})}$$

$$\times(-1)^{|\mathcal{M}|}\left(\prod_{(i,j)\in\mathcal{M}}C_{ij}\right)\left(\prod_{i\in X\setminus X_{\boldsymbol{m}}}F_i\right)\left(\prod_{i\in X_{\boldsymbol{m}}\setminus X}E_i\right)\left(\prod_{i\in X\cap X_{\boldsymbol{m}}}D_i\right)\left(\prod_{i\in Y\setminus A(\mathcal{M})\setminus X}A_i\right)\left(\prod_{i\in Y^c\setminus B(\mathcal{M})}B_i\right) \tag{F44}$$

$$=\sum_{Y\subset X_{\boldsymbol{n}}}\sum_{\mathcal{M}\in\mathrm{Match}(F_{X_{\boldsymbol{n}}};Y\to Y^c)}\prod_{(i,j)\in\mathcal{M}}(-C_{ij})\prod_{i\in Y\setminus A(\mathcal{M})\setminus X_{\boldsymbol{m}}}(A_i+F_i)\prod_{i\in X_m}(E_iA_i+D_i)\prod_{i\in Y^c\setminus B(\mathcal{M})}B_i \tag{F45}$$

$$=\prod_{i\in X_m}(E_iA_i+D_i)\mathrm{lhaf}(\tilde{K}), \tag{F46}$$

where $\tilde{K}$ is a $|X_{\boldsymbol{n}}| \times |X_{\boldsymbol{n}}|$ matrix whose elements are given as

$$\tilde{K}_{ij} = \begin{cases} -C_{ij} & \text{if } i < j \text{ and} \\ A_i + B_i + F_i & \text{if } i = j \text{ and } i \notin X_{\boldsymbol{m}}, \\ B_i + 1 & \text{if } i = j \text{ and } i \in X_{\boldsymbol{m}}, \end{cases} \tag{F47}$$

where $i, j \in X_{\boldsymbol{n}}$. Hence, the low-rank structure of C may be used. Let us also define a matrix K as the symmetrized version of $\tilde{K}$, i.e.,

$$K_{ij} = \begin{cases} -C_{ij} & \text{if } i \neq j \text{ and} \\ A_i + B_i + F_i & \text{if } i = j \text{ and } i \notin X_{\boldsymbol{m}}, \\ B_i + 1 & \text{if } i = j \text{ and } i \in X_{\boldsymbol{m}}, \end{cases} \tag{F48}$$

and note that $\mathrm{lhaf}(K) = \mathrm{lhaf}(\tilde{K})$ if we define the loop hafnian to depend only on the upper triangular matrix (see Appendix G). To exploit this low-rank structure, we can then invoke the following lemma:

Lemma F2 (Loop hafnian of a matrix with low-rank structure [64]). Let Σ be an $n \times n$ symmetric matrix with rank r and A be a matrix constructed by filling Σ's diagonal matrix with arbitrary elements. Then the loop hafnian of the matrix A can be computed in $O(n\binom{2n+r-1}{r-1})$. Thus, when r is fixed, the cost is polynomial in n and when $r = O(\log n)$, the cost is quasipolynomial in n.

As an example, let us first consider the case that $\boldsymbol{n} = \boldsymbol{m} = (1, \ldots, 1)$, i.e., $X_{\boldsymbol{n}} = X_{\boldsymbol{m}} = [M]$. Then, the matrix K can be treated as a matrix constructed by replacing the diagonal matrix of matrix C with $B_i + 1$. Hence, we can use the lemma, and the complexity is $O(M\binom{2M+2L-1}{2L-1})$. (See Appendix G for the algorithm.) More specifically, since C has a rank $2L$, we find a decomposition $C = GG^{\mathrm{T}}$, where G is an $M \times 2L$ matrix and then construct the polynomial

$$g(y_1, \ldots, y_{2L}) = \prod_{i=1}^{M} \left(\sum_{j=1}^{2L} G_{ij} y_j + B_i + 1 \right) (E_i A_i + D_i), \tag{F49}$$

and multiplying appropriate prefactors to obtain the loop hafnian.

Now, let us consider more general $\boldsymbol{n}$ and $\boldsymbol{m}$. In this case, the matrix K is constructed by repeating rows and columns of the matrix C according to $X_{\boldsymbol{n}}$ and filling the matrix's diagonal elements according to the definition of K. Hence, this matrix still has a low-rank structure that enables us to invoke the above lemma. One may then observe that if we let $l_i \equiv \min(n_i, m_i)$, define a polynomial

$$g(y_1, \ldots, y_{2L}) = \frac{1}{\sqrt{\boldsymbol{n}!\boldsymbol{m}!}} \prod_{i=1}^{M} \left[\left(\sum_{j=1}^{2L} G_{ij} y_j + B_i + 1 \right)^{l_i} \left(\sum_{j=1}^{2L} G_{ij} y_j + A_i + B_i + F_i \right)^{\max(0, n_i - l_i)} (E_i A_i + D_i)^{m_i} \right], \tag{F50}$$

and multiply appropriate prefactors, we obtain the loop hafnian. The final expression is a polynomial of $\{x_i\}_{i=1}^{2L}$ and $\{y_i\}_{i=1}^{2L}$ and $\{a_u^{(i)}\}_{i=1}^{2L}$, $\{a_v^{(i)}\}_{i=1}^{2L}$, $\{a_w^{(i)}\}_{i=1}^{2L}$, $\{a_x^{(i)}\}_{i=1}^{2L}$, $\{a_y^{(i)}\}_{i=1}^{2L}$, and $\{a_z^{(i)}\}_{i=1}^{2L}$. Finally, to deal with the squeezed states, we replace x_i with the corresponding coefficients of the squeezed states. To be more explicit, x_i is replaced with $\cosh^{-1/2} r \tanh^{i/2} r \sqrt{i!}/(2^{i/2}(i/2)!)$ for even i and with 0 for odd i, where r is the squeezing parameter for each. Then, y_i's are taken care of using the method in Appendix G, and $\{a_u^{(i)}\}_{i=1}^{2L}$, $\{a_v^{(i)}\}_{i=1}^{2L}$, $\{a_w^{(i)}\}_{i=1}^{2L}$, $\{a_x^{(i)}\}_{i=1}^{2L}$, $\{a_y^{(i)}\}_{i=1}^{2L}$, and $\{a_z^{(i)}\}_{i=1}^{2L}$ are taken care of as explained above.

Finally, let us now consider the general case,

$$\langle \phi | \hat{G} | \psi \rangle = \sum_{\boldsymbol{n}, \boldsymbol{m}} a_{\boldsymbol{m}} b_{\boldsymbol{n}}^* \langle \boldsymbol{n} | \hat{G} | \boldsymbol{m} \rangle, \tag{F51}$$

which is the goal of this Appendix. Then, we define a polynomial

$$g(y_1,\ldots,y_{2L}) = \prod_{i=1}^{M}\sum_{n_i,m_i=0}^{n_{\max}} \frac{a_{m_i}^{(i)} b_{n_i}^{(i)*}}{\sqrt{n_i!m_i!}} \left[\left(\sum_{j=1}^{2L} G_{ij}y_j + B_i + 1\right)^{l_i} \left(\sum_{j=1}^{2L} G_{ij}y_j + A_i + B_i + F_i\right)^{\max(0,n_i-l_i)} (E_iA_i + D_i)^{m_i}\right]. \tag{F52}$$

Then, by multiplying appropriate prefactors as above for squeezed states' coefficients, we obtain $\langle\phi|\hat{G}|\psi\rangle$ by appropriately selecting coefficients of the monomials as above.

Let us now analyze the complexity. Note that we have introduced indeterminates $\{x_i\}_{i=1}^{2L}$ and $\{y_i\}_{i=1}^{2L}$ and $\{a_u^{(i)}\}_{i=1}^{2L}$, $\{a_v^{(i)}\}_{i=1}^{2L}$, $\{a_w^{(i)}\}_{i=1}^{2L}$, $\{a_x^{(i)}\}_{i=1}^{2L}$, $\{a_y^{(i)}\}_{i=1}^{2L}$, and $\{a_z^{(i)}\}_{i=1}^{2L}$. Thus, the number of indeterminates we introduced is $16L$. All these indeterminates can have a degree at most $3Mn_{\max}$ due to the quadratic form of E_iA_i in the expression and $D_i = 1 + f_2(i)(f_4(i) + g_2(i))$. Thus, the number of coefficients we need to keep track of is $(3Mn_{\max}+1)^{16L}$. Therefore, when $L = O(1)$, it can be computed in polynomial time. ■

APPENDIX G: LOW-RANK LOOP HAFNIAN

For completeness, we provide the details of a classical algorithm that computes the loop hafnian of a matrix with a low-rank structure, which was proposed in Ref. [64] by generalizing the algorithm for hafnian [63].

Let us first recall the definition of the loop hafnian of an $n \times n$ matrix A. Consider the graph G whose adjacency matrix is A. Here, the diagonal elements of A represent the loops from a vertex to itself. Then, the loop hafnian can be written as

$$\mathrm{lhaf}(A) \equiv \sum_{\mathcal{M}\in \mathrm{SPM}} \prod_{(i,j)\in\mathcal{M}} A_{ij}, \tag{G1}$$

where SPM is the set of all possible matchings of a graph G, including loops. In many cases, as the hafnian, we assume A to be a symmetric matrix because of the undirectedness of the graph. However, one may define the loop hafnian of a matrix that is not symmetric by always considering a match (i,j) as either $i \le j$ or $i \ge j$. Then, the loop hafnian depends only on the upper or lower triangular part, respectively. This property was used in Ref. [63] and is used below to employ the low-rank structure for computing the loop hafnian.

Before we present the algorithm, let us define the loop hafnian in a different way. Let $X \equiv [n]$. Then, the loop hafnian can be written as

$$\mathrm{lhaf}(A) = \sum_{Y\subset X} \sum_{\mathcal{M}\in\mathrm{Match}(F_X;Y\to Y^c)} \prod_{(i,j)\in\mathcal{M}} A_{ij} \prod_{i\in Y\setminus A(\mathcal{M})} A_{ii}/2 \times \prod_{i\in Y^c\setminus B(\mathcal{M})} A_{ii}/2. \tag{G2}$$

Here, F_X is the Ferrers graph, i.e., its edges are $E = \{(i,j) : j < i\}$ and $1/2$ factor is introduced to cancel out the redundancy, and $A(\mathcal{M})$ and $B(\mathcal{M})$ represent the elements in Y and Y^c that are involved in the matching $\mathcal{M}$, respectively. Hence, this is consistent with the argument regarding the lower triangular matrix above. Thus, the first product corresponds to the matches between distinct vertices, and the last two products correspond to the loops. We may generalize this expression to

$$\sum_{Y\subset X} \sum_{\mathcal{M}\in\mathrm{Match}(F_X;Y\to Y^c)} \prod_{(i,j)\in\mathcal{M}} C_{ij} \prod_{i\in Y\setminus A(\mathcal{M})} A_i \prod_{i\in Y^c\setminus B(\mathcal{M})} B_i, \tag{G3}$$

and this is equivalent to

$$\mathrm{lhaf}(W), \tag{G4}$$

where the matrix W's off-diagonal elements are equal to C_{ij} and diagonal elements are $A_i + B_i$.

We now provide the classical algorithm that takes advantage of the low-rank structure [64]. Consider a symmetric matrix Σ and another symmetric matrix A which is obtained by replacing Σ's diagonal elements by μ. When Σ has rank r, it can be decomposed as $\Sigma = GG^{\mathrm{T}}$, where G is an $n \times r$ matrix. Then, we can compute the loop hafnian of the matrix as

$$\mathrm{lhaf}(A) = \sum_{\mathcal{M}\in\mathrm{SPM}} \prod_{(i,j)\in\mathcal{M}} A_{ij} = \sum_{a\subset[n]:|a|:\mathrm{even}} \mathrm{haf}(A_{a,a}) \prod_{i\in a^c} \mu_i$$

$$= \sum_{\sigma\in\mathrm{PMP}(|a|)} \prod_{i=1}^{|a|} A_{a(\sigma(2i-1)),a(\sigma(2i))} \prod_{i\in a^c} \mu_i \tag{G5}$$

$$\to \prod_{i=1}^{n}\left(\sum_{j=1}^{r} G_{ij}x_j + \mu_i\right), \tag{G6}$$

where PMP is the set of perfect matchings without loops, and we introduced a polynomial with indeterminates x_i's. Let $\mathcal{P}_{2s,r}$ be the set of integer r-partitions of $2s$, i.e., tuples $(p_1,p_2,\ldots,p_r)$ such that $p_i \ge 0$ for all i and $\sum_i p_i = 2s$, and let $\mathcal{E}_{2s,r}$ be the subset of $\mathcal{P}_{s,r}$ such that p_i is even for all i. We then expand the polynomial and identify the

coefficients λ_p as

$$q(x_1,\ldots,x_r)=\sum_{s=0}^{[n/2]}\sum_{p=(p_1,\ldots,p_{2s})\in\mathcal{P}_{2s,r}}\lambda_p\prod_{i=1}^{r}x_i^{p_i}, \tag{G7}$$

where we dropped a polynomial composed of an odd number of x_i's from the polynomial, Eq. (G6), because it is not relevant to the loop hafnian. Since

$$|\mathcal{P}_{2s,r}|=\binom{2s+r-1}{r-1},\quad |\mathcal{E}_{2s,r}|=\binom{s+r-1}{r-1}, \tag{G8}$$

the coefficients λ_p can be found in $O(n|\mathcal{P}_{2n,r}|)$ running time because there are $|\mathcal{P}_{2n,r}|$ monomials whose coefficients are kept track of and we have n multiplications. Meanwhile, the loop hafnian can be written as

$$\mathrm{lhaf}(A)=\sum_{s=0}^{[n/2]}\sum_{e\in\mathcal{E}_{2s,r}}\lambda_e\prod_{i=1}^{r}(e_i-1)!!. \tag{G9}$$

Therefore, by expanding the polynomial (G7) and identifying the coefficients, we can compute the loop hafnian in $O(n|\mathcal{P}_{2n,r}|)$ and when r is fixed, in $O(n^r)$.

[1] P. W. Shor, in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (IEEE, Los Alamitos, CA, 1994), pp. 124–134.

[2] S. Lloyd, Universal quantum simulators, Science **273**, 1073 (1996).

[3] A. Bouland, B. Fefferman, C. Nirkhe, and U. Vazirani, On the complexity and verification of quantum random circuit sampling, Nat. Phys. **15**, 159 (2019).

[4] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell *et al.*, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[5] Y. Wu, W.-S. Bao, S. Cao, F. Chen, M.-C. Chen, X. Chen, T.-H. Chung, H. Deng, Y. Du, D. Fan *et al.*, Strong quantum computational advantage using a superconducting quantum processor, Phys. Rev. Lett. **127**, 180501 (2021).

[6] A. Morvan, B. Villalonga, X. Mi, S. Mandrà, A. Bengtsson, P. Klimov, Z. Chen, S. Hong, C. Erickson, I. Drozdov *et al.*, Phase transition in random circuit sampling, Nature **634**, 328 (2024).

[7] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu *et al.*, Quantum computational advantage using photons, Science **370**, 1460 (2020).

[8] H.-S. Zhong, Y.-H. Deng, J. Qin, H. Wang, M.-C. Chen, L.-C. Peng, Y.-H. Luo, D. Wu, S.-Q. Gong, H. Su *et al.*, Phase-programmable Gaussian boson sampling using stimulated squeezed light, Phys. Rev. Lett. **127**, 180502 (2021).

[9] L. S. Madsen, F. Laudenbach, M. F. Askarani, F. Rortais, T. Vincent, J. F. Bulmer, F. M. Miatto, L. Neuhaus, L. G. Helt, M. J. Collins *et al.*, Quantum computational advantage with a programmable photonic processor, Nature **606**, 75 (2022).

[10] Y.-H. Deng, Y.-C. Gu, H.-L. Liu, S.-Q. Gong, H. Su, Z.-J. Zhang, H.-Y. Tang, M.-H. Jia, J.-M. Xu, M.-C. Chen *et al.*, Gaussian boson sampling with pseudo-photon-number-resolving detectors and quantum computational advantage, Phys. Rev. Lett. **131**, 150601 (2023).

[11] A. W. Young, S. Geller, W. J. Eckner, N. Schine, S. Glancy, E. Knill, and A. M. Kaufman, An atomic boson sampler, Nature **629**, 311 (2024).

[12] H.-L. Liu, H. Su, S.-Q. Gong, Y.-C. Gu, H.-Y. Tang, M.-H. Jia, Q. Wei, Y. Song, D. Wang, M. Zheng *et al.*, Robust quantum computational advantage with programmable 3050-photon Gaussian boson sampling, arXiv:2508.09092.

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

[14] P. Clifford and R. Clifford, in *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms* (SIAM, Philadelphia, PA, 2018), pp. 146–155.

[15] M. Oszmaniec and D. J. Brod, Classical simulation of photonic linear optics with lost particles, New J. Phys. **20**, 092002 (2018).

[16] R. García-Patrón, J. J. Renema, and V. Shchesnovich, Simulating boson sampling in lossy architectures, Quantum **3**, 169 (2019).

[17] H. Qi, D. J. Brod, N. Quesada, and R. García-Patrón, Regimes of classical simulability for noisy Gaussian boson sampling, Phys. Rev. Lett. **124**, 100502 (2020).

[18] C. Oh, K. Noh, B. Fefferman, and L. Jiang, Classical simulation of lossy boson sampling using matrix product operators, Phys. Rev. A **104**, 022407 (2021).

[19] N. Quesada, R. S. Chadwick, B. A. Bell, J. M. Arrazola, T. Vincent, H. Qi, R. García *et al.*, Quadratic speed-up for simulating Gaussian boson sampling, PRX Quantum **3**, 010306 (2022).

[20] J. F. Bulmer, B. A. Bell, R. S. Chadwick, A. E. Jones, D. Moise, A. Rigazzi, J. Thorbecke, U.-U. Haus, T. Van Vaerenbergh, R. B. Patel *et al.*, The boundary for quantum advantage in Gaussian boson sampling, Sci. Adv. **8**, eabl9236 (2022).

[21] C. Oh, Y. Lim, B. Fefferman, and L. Jiang, Classical simulation of boson sampling based on graph structure, Phys. Rev. Lett. **128**, 190501 (2022).

[22] C. Oh, L. Jiang, and B. Fefferman, On classical simulation algorithms for noisy boson sampling, arXiv:2301.11532.

[23] M. Liu, C. Oh, J. Liu, L. Jiang, and Y. Alexeev, Simulating lossy Gaussian boson sampling with matrix-product operators, Phys. Rev. A **108**, 052604 (2023).

[24] C. Oh, M. Liu, Y. Alexeev, B. Fefferman, and L. Jiang, Classical algorithm for simulating experimental Gaussian boson sampling, Nat. Phys. **20**, 1461 (2024).

[25] C. Oh, Recent theoretical and experimental progress on boson sampling, Curr. Opt. Photonics **9**, 1 (2025).

[26] C. Oh, Classical simulability of constant-depth linear-optical circuits with noise, npj Quantum Inf. **11**, 126 (2025).

[27] S. Aaronson and A. Arkhipov, in *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 2011), pp. 333–342.

[28] C. S. Hamilton, R. Kruse, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex, Gaussian boson sampling, Phys. Rev. Lett. **119**, 170501 (2017).
[29] A. Deshpande, A. Mehta, T. Vincent, N. Quesada, M. Hinsche, M. Ioannou, L. Madsen, J. Lavoie, H. Qi, J. Eisert, D. Hangleiter, B. Fefferman, and I. Dhand, Quantum computational advantage via high-dimensional Gaussian boson sampling, Sci. Adv. **8**, eabi7894 (2022).
[30] E. Knill, R. Laflamme, and G. J. Milburn, A scheme for efficient quantum computation with linear optics, Nature **409**, 46 (2001).
[31] A. Mari and J. Eisert, Positive Wigner functions render classical simulation of quantum computation efficient, Phys. Rev. Lett. **109**, 230503 (2012).
[32] C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, Gaussian quantum information, Rev. Mod. Phys. **84**, 621 (2012).
[33] S. Rahimi-Keshari, T. C. Ralph, and C. M. Caves, Sufficient conditions for efficient classical simulation of quantum optics, Phys. Rev. X **6**, 021039 (2016).
[34] B. Q. Baragiola, G. Pantaleoni, R. N. Alexander, A. Karanjai, and N. C. Menicucci, All-Gaussian universality and fault tolerance with the Gottesman-Kitaev-Preskill code, Phys. Rev. Lett. **123**, 200502 (2019).
[35] S. Lloyd and S. L. Braunstein, Quantum computation over continuous variables, Phys. Rev. Lett. **82**, 1784 (1999).
[36] Ch. Silberhorn, P. K. Lam, O. Weiss, F. König, N. Korolkova, and G. Leuchs, Generation of continuous variable Einstein-Podolsky-Rosen entanglement via the Kerr nonlinearity in an optical fiber, Phys. Rev. Lett. **86**, 4267 (2001).
[37] S. Scheel, K. Nemoto, W. J. Munro, and P. L. Knight, Measurement-induced nonlinearity in linear optics, Phys. Rev. A **68**, 032310 (2003).
[38] J. M. Arrazola, V. Bergholm, K. Brádler, T. R. Bromley, M. J. Collins, I. Dhand, A. Fumagalli, T. Gerrits, A. Goussev, L. G. Helt *et al.*, Quantum circuits with many photons on a programmable nanophotonic chip, Nature **591**, 54 (2021).
[39] J. E. Bourassa, R. N. Alexander, M. Vasmer, A. Patil, I. Tzitrin, T. Matsuura, D. Su, B. Q. Baragiola, S. Guha, G. Dauphinais *et al.*, Blueprint for a scalable photonic fault-tolerant quantum computer, Quantum **5**, 392 (2021).
[40] N. Spagnolo, D. J. Brod, E. F. Galvão, and F. Sciarrino, Non-linear boson sampling, npj Quantum Inf. **9**, 3 (2023).
[41] M. V. Larsen, J. E. Bourassa, S. Kocsis, J. F. Tasker, R. S. Chadwick, C. González-Arciniegas, J. Hastrup, C. E. Lopetegui-González, F. M. Miatto, A. Motamedi *et al.*, Integrated photonic source of Gottesman–Kitaev–Preskill qubits, Nature **642**, 587 (2025).
[42] H. Aghaee Rad, T. Ainsworth, R. N. Alexander, B. Altieri, M. F. Askarani, R. Baby, L. Banchi, B. Q. Baragiola, J. E. Bourassa, R. Chadwick *et al.*, Scaling and networking a modular photonic quantum computer, Nature **638**, 912 (2025).
[43] K. Takase, K. Fukui, A. Kawasaki, W. Asavanant, M. Endo, J.-i. Yoshikawa, P. van Loock, and A. Furusawa, Gottesman–Kitaev–Preskill qubit synthesizer for propagating light, npj Quantum Inf. **9**, 72 (2023).
[44] U. Chabaud, D. Markham, and A. Sohbi, Quantum machine learning with adaptive linear optics, Quantum **5**, 496 (2021).
[45] F. Hoch, E. Caruccio, G. Rodari, T. Francalanci, A. Suprano, T. Giordani, G. Carvacho, N. Spagnolo, S. Koudia, M. Proietti *et al.*, Quantum machine learning with adaptive boson sampling via post-selection, Nat. Commun. **16**, 902 (2025).
[46] S. Bravyi, D. Gosset, and R. Movassagh, Classical algorithms for quantum mean values, Nat. Phys. **17**, 337 (2021).
[47] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. van den Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme *et al.*, Evidence for the utility of quantum computing before fault tolerance, Nature **618**, 500 (2023).
[48] Z. Yin, I. Agresti, G. de Felice, D. Brown, A. Toumi, C. Pentangelo, S. Piacentini, A. Crespi, F. Ceccarelli, R. Osellame *et al.*, Experimental quantum-enhanced kernels on a photonic processor, arXiv:2407.20364.
[49] E. Bernstein and U. Vazirani, in *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 1993), pp. 11–20.
[50] A. Y. Kitaev, A. Shen, and M. N. Vyalyi, *Classical and Quantum Computation* (American Mathematical Society, Providence, RI, 2002), Vol. 47.
[51] H. Pashayan, S. D. Bartlett, and D. Gross, From estimation of quantum probabilities to simulation of quantum circuits, Quantum **4**, 223 (2020).
[52] D. Hangleiter and J. Eisert, Computational advantage of quantum random sampling, Rev. Mod. Phys. **95**, 035001 (2023).
[53] U. Chabaud and M. Walschaers, Resources for bosonic quantum computational advantage, Phys. Rev. Lett. **130**, 090602 (2023).
[54] O. Hahn, R. Takagi, G. Ferrini, and H. Yamasaki, Classical simulation and quantum resource theory of non-Gaussian optics, Quantum **9**, 1881 (2025).
[55] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. **5**, 4213 (2014).
[56] A. Pappalardo, P.-E. Emeriau, G. de Felice, B. Ventura, H. Jaunin, R. Yeung, B. Coecke, and S. Mansfield, A photonic parameter-shift rule: Enabling gradient computation for photonic quantum computers, Phys. Rev. A **111**, 032429 (2025).
[57] F. Hoch, G. Rodari, T. Giordani, P. Perret, N. Spagnolo, G. Carvacho, C. Pentangelo, S. Piacentini, A. Crespi, F. Ceccarelli *et al.*, Variational approach to photonic quantum circuits via the parameter shift rule, Phys. Rev. Research **7**, 023227 (2025).
[58] G. Facelli, D. D. Roberts, H. Wallner, A. Makarovskiy, Z. Holmes, and W. R. Clements, Exact gradients for linear optics with single photons, arXiv:2409.16369.
[59] A. Baldazzi, M. Sanna, M. Borghi, S. Azzini and L. Pavesi, Four-qubit variational quantum algorithms in silicon photonics with integrated entangled photon sources, npj Quantum Inf. **11**, 107 (2025).
[60] N. Maring, A. Fyrillas, M. Pont, E. Ivanov, P. Stepanov, N. Margaria, W. Hease, A. Pishchagin, A. Lemaître, I. Sagnes *et al.*, A versatile single-photon-based quantum computing platform, Nat. Photonics **18**, 603 (2024).
[61] D. A. Ivanov and L. Gurvits, Complexity of full counting statistics of free quantum particles in product states, Phys. Rev. A **101**, 012303 (2020).

[62] A. I. Barvinok, Two algorithmic results for the traveling salesman problem, Math. Oper. Res. **21**, 65 (1996).
[63] A. Björklund, B. Gupt, and N. Quesada, A faster hafnian formula for complex matrices and its benchmarking on a supercomputer, J. Exp. Algorithm. (JEA) **24**, 1 (2019).
[64] C. Oh, Y. Lim, Y. Wong, B. Fefferman, and L. Jiang, Quantum-inspired classical algorithms for molecular vibronic spectra, Nat. Phys. **20**, 225 (2024).
[65] S. Aaronson and D. Gottesman, Improved simulation of stabilizer circuits, Phys. Rev. A **70**, 052328 (2004).
[66] S. Bravyi and D. Gosset, Improved classical simulation of quantum circuits dominated by Clifford gates, Phys. Rev. Lett. **116**, 250501 (2016).
[67] D. Gottesman, in *Group22: Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics,* edited by S. P. Corney, R. Delbourgo, and P. D. Jarvis (International Press, Cambridge, MA, 1999), pp. 32–43.
[68] V. Veitch, C. Ferrie, D. Gross, and J. Emerson, Negative quasi-probability as a resource for quantum computation, New J. Phys. **14**, 113011 (2012).
[69] U. Chabaud, G. Ferrini, F. Grosshans, and D. Markham, Classical simulation of Gaussian quantum circuits with non-Gaussian input states, Phys. Rev. Res. **3**, 033018(2021).
[70] J. E. Bourassa, N. Quesada, I. Tzitrin, A. Száva, T. Isacsson, J. Izaac, K. K. Sabapathy, G. Dauphinais, and I. Dhand, Fast simulation of bosonic qubits via Gaussian functions in phase space, PRX Quantum **2**, 040315 (2021).
[71] J. Marshall and N. Anand, Simulation of quantum optics by coherent state decomposition, Opt. Quantum **1**, 78 (2023).
[72] B. Dias and R. König, Classical simulation of non-Gaussian bosonic circuits, Phys. Rev. A **110**, 042402 (2024).
[73] C. Calcluth, N. Reichel, A. Ferraro, and G. Ferrini, Sufficient condition for universal quantum computation using bosonic circuits, PRX Quantum **5**, 020337 (2024).
[74] F. A. Mele, S. F. E. Oliviero, V. Upreti, and U. Chabaud, The symplectic rank of non-Gaussian quantum states, arXiv:2504.19319.
[75] H. Pashayan, J. J. Wallman, and S. D. Bartlett, Estimating outcome probabilities of quantum circuits using quasiprobabilities, Phys. Rev. Lett. **115**, 070501 (2015).
[76] Y. Lim and C. Oh, Approximating outcome probabilities of linear optical circuits, npj Quantum Inf. **9**, 124 (2023).
[77] Y. Lim and C. Oh, Efficient classical algorithms for linear optical circuits, arXiv:2502.12882.
[78] U. Chabaud, R. Ghobadi, S. Beigi, and S. Rahimi-Keshari, Phase-space negativity as a computational resource for quantum Kernel methods, Quantum **8**, 1519 (2024).
[79] K. E. Cahill and R. J. Glauber, Density operators and quasiprobability distributions, Phys. Rev. **177**, 1882 (1969).
[80] A. Ferraro, S. Olivares, and M. G. Paris, Gaussian states in continuous variable quantum information, arXiv:quant-ph/0503237.
[81] A. Serafini, *Quantum Continuous Variables: a Primer of Theoretical Methods* (CRC, Boca Raton, FL, 2017).
[82] M. R. Jerrum, L. G. Valiant, and V. V. Vazirani, Random generation of combinatorial structures from a uniform distribution, Theor. Comput. Sci. **43**, 169 (1986).
[83] We note that when marginal probabilities can be efficiently computed, sampling from each mode is efficient because the only scaling parameter, defining the efficiency, comes from the marginal probabilities. However, there may be additional costs in practice for sampling, especially for continuous variables (e.g., rejection sampling).
[84] L. Chakhmakhchyan and N. J. Cerf, Boson sampling with Gaussian measurements, Phys. Rev. A **96**, 032326 (2017).
[85] S. D. Bartlett, B. C. Sanders, S. L. Braunstein, and K. Nemoto, Efficient classical simulation of continuous variable quantum information processes, Phys. Rev. Lett. **88**, 097904 (2002).
[86] C. P. Robert, G. Casella, and G. Casella, *Monte Carlo Statistical Methods* (Springer, 1999), Vol. 2.
[87] M. Fukuda and R. Koenig, Typical entanglement for Gaussian states, J. Math. Phys. **60**, 112203 (2019).
[88] J. T. Iosue, A. Ehrenberg, D. Hangleiter, A. Deshpande, and A. V. Gorshkov, Page curves and typical entanglement in linear optics, Quantum **7**, 1017 (2023).
[89] U. Chabaud, D. Markham, and F. Grosshans, Stellar representation of non-Gaussian quantum states, Phys. Rev. Lett. **124**, 063605 (2020).
[90] N. Quesada, Franck-Condon factors by counting perfect matchings of graphs with loops, J. Chem. Phys. **150**, 164113 (2019).
[91] H. J. Ryser, *Combinatorial Mathematics* (American Mathematical Society, Washington, DC, 1963), Vol. 14.
[92] A. Varvak, Rook numbers and the normal ordering problem, J. Comb. Theory, Ser. A **112**, 292 (2005).
[93] R. P. Stanley, *Enumerative Combinatorics, Volume 1* (Cambridge University, Cambridge, 2011), Vol. 49.

Correction: The author list in Ref. [59] contained errors and has been fixed.