


Continuous-variable quantum approximate optimization on a programmable photonic quantum processor

Yutaro Enomoto,^{*} Keitaro Anai, Kenta Udagawa, and Shuntaro Takeda[†]

Department of Applied Physics, School of Engineering, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

 (Received 2 June 2023; revised 1 August 2023; accepted 5 September 2023; published 3 October 2023)

Variational quantum algorithms (VQAs) provide a promising approach to achieving quantum advantage for practical problems on near-term noisy intermediate-scale quantum (NISQ) devices. Thus far, most studies on VQAs have focused on qubit-based systems, but the power of VQAs can be potentially boosted by exploiting infinite-dimensional continuous-variable (CV) systems. Here, we implement the CV version of one VQA, a quantum approximate optimization algorithm, by developing an automated collaborative computing system between a programmable photonic quantum computer and a classical computer. We experimentally demonstrate that this algorithm solves the minimization problem of simple continuous functions by implementing the quantum version of gradient descent to localize an initially broadly distributed wave function to the minimum. This method allows the execution of a practical CV quantum algorithm on a physical platform. Our work can be extended to the minimization of more general functions, providing an alternative to achieve the quantum advantage in practical problems.

DOI: [10.1103/PhysRevResearch.5.043005](https://doi.org/10.1103/PhysRevResearch.5.043005)

I. INTRODUCTION

Variational quantum algorithms (VQAs) have recently emerged as the leading approach to achieving quantum advantage for practical problems under the constraints of near-term noisy intermediate-scale quantum (NISQ) devices [1,2]. In VQAs, such constraints are avoided by the common strategy to repeatedly run shallow-depth quantum circuits with the circuit parameters updated by classical optimizers. This strategy enables us to mitigate the accumulation of errors and fully exploit the computational space offered by the limited-scale devices. Thus far, a wide variety of VQAs have been proposed theoretically for qubit-based systems, such as ones for combinatorial optimization [3], chemistry simulation [4], and machine learning [5]. They have already been demonstrated experimentally on several physical platforms [4,6–10].

In contrast, there have been much fewer proposals [11–15] and no experimental implementations on continuous-variable (CV) VQAs, although CV quantum computing can potentially offer superior computational power in the NISQ era. The potential of CV systems lies in the ability to process infinite-dimensional quantum information even on single-mode devices, while in qubit-based systems each qubit provides only two-dimensional computational space. Furthermore, CV systems natively and efficiently handle continuous real parameters that often appear in real-world problems. In general, fully exploiting such infinite dimensionality and

continuous degree of freedom in CV systems for quantum computation has been regarded as impractical due to their noise sensitivity and difficulty in error correction [16]. However, this can be in turn a promising approach to extracting high computational power in the NISQ era when the error correction is not assumed.

In this paper, we implement the CV version [11] of one of the most typical VQAs, a quantum approximate optimization algorithm (QAOA) [3], by developing an automated collaborative computing system between a programmable photonic quantum computer and a classical computer. We choose the simplest problem for the CV-QAOA and experimentally demonstrate it, where this algorithm minimizes one-variable continuous quadratic functions by working as the quantum version of gradient descent and localizing an initially broadly distributed wave function to the minimum. The algorithm is shown to robustly find approximate answers to the problem with a noisy shallow-depth quantum circuit, thus confirming that a VQA works also in CV systems. This method showcases a CV quantum algorithm that can be used to solve practical problems, except for Gaussian boson sampling [17–19], which is designed for achieving quantum supremacy and has been partially linked to some practical problems [20,21]. We also show that our implementation can be extended to the minimization problems of arbitrary-order functions by adding optical resources [22]. It can also be extended to multivariable functions by incorporating multimode interactions. Thus our work highlights an approach using CV quantum computing in the NISQ era, offering an alternative to achieve the quantum advantage in practical problems.

II. THEORY OF CV-QAOA

Many practical problems in various fields come down to optimization or minimization, which often require high computational costs for classical computers. The QAOA is

^{*}yenomoto.ap.t@gmail.com

[†]takeda@ap.t.u-tokyo.ac.jp

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.

a heuristic algorithm that could potentially offer a quantum speedup to solve such problems on NISQ devices [3]. Theoretical aspects and experimental implementations of the QAOA have been recently studied intensively on qubit-based systems to solve discrete combinatorial optimization problems [6–9,23,24].

Later, a CV version of the QAOA was proposed to solve continuous optimization problems on CV systems [11]. This algorithm is designed for minimizations of continuous real-valued functions, which have many practical applications in finance [25], machine learning [26], and engineering [27]. This proposal [11] indicates that the CV-QAOA has the potential of a quantum speedup as its circuits can encode CV Grover's search algorithm [28], which achieves a quadratic speedup over the classical algorithms.

The CV-QAOA is formulated as follows. The goal of the algorithm is to find an approximate minimum of a real-valued function $f(\mathbf{x})$ with $\mathbf{x} = (x_1, x_2, \dots, x_N) \in \mathbb{R}^N$. Let us consider a quantum-mechanical particle in the N -dimensional space with $[\hat{x}_i, \hat{p}_j] = i\delta_{ij}$, where $(\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N) = \hat{\mathbf{x}}$ and $(\hat{p}_1, \hat{p}_2, \dots, \hat{p}_N) = \hat{\mathbf{p}}$ are position and momentum of the particle, respectively. The initial state is $|\mathbf{p} = 0\rangle$, which is an eigenstate of $\hat{\mathbf{p}}$ and thus equally weighted superposition of $|\mathbf{x}\rangle$ for all \mathbf{x} . The unitary operator given by

$$\hat{U}(\boldsymbol{\eta}, \boldsymbol{\gamma}) = \prod_{j=1}^P e^{-i\gamma_j \hat{H}_M} e^{-i\eta_j \hat{H}_C}, \quad (1)$$

where $\hat{H}_C = f(\hat{\mathbf{x}})$ and $\hat{H}_M = \hat{\mathbf{p}}^2/2$, transforms the initial state to the final state. P determines the circuit depth. \hat{H}_C and \hat{H}_M are called cost and mixer Hamiltonians, respectively, and $\boldsymbol{\eta} = (\eta_1, \eta_2, \dots, \eta_P)$ and $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_P)$ are the tunable real positive parameters. The final state is then measured in the $\hat{\mathbf{x}}$ basis, the measurement outcome \mathbf{x} being a candidate of the minimum of the function. This algorithm can be considered as a quantum version of the gradient descent method; a pair of the cost and mixer Hamiltonians transforms $\hat{\mathbf{x}}$ as

$$\hat{\mathbf{x}} \rightarrow \hat{\mathbf{x}} + \gamma_j \hat{\mathbf{p}} - \eta_j \nabla f(\hat{\mathbf{x}}) \quad (2)$$

and indeed $\hat{U}(\boldsymbol{\eta}, \boldsymbol{\gamma})$ represents the Trotterized approximation of the time evolution of a particle trapped in a potential of $f(\hat{\mathbf{x}})$ in an N -dimensional space [3,11]. Thus, while the distribution of \mathbf{x} is initially uniform, it moves under the influence of the potential $f(\hat{\mathbf{x}})$ and then localizes around the minimum if the parameters $(\boldsymbol{\eta}, \boldsymbol{\gamma})$ are properly chosen.

For the demonstration of the algorithm, we adopt the simplest problem setup, minimizing a quadratic function of one variable with the shallowest depth. Specifically, we choose $f(x) = (x - a)^2$ with $a \in \mathbb{R}$ ($N = 1$) and set $P = 1$.

III. PHOTONIC-CIRCUIT IMPLEMENTATION

We implement the CV-QAOA on a programmable photonic quantum computer, where optical amplitude and phase are identified with position and momentum in the algorithm, respectively. Our implementation is enabled by recent technological advances in photonic CV quantum computing, including state preparations, gate operations, and measurements [29]. Specifically, programmable and multistep CV gate operations have been demonstrated very recently, which are

indispensable for implementing the CV-QAOA [30–32]. Such previous studies have been limited to the proof-of-principle demonstrations of predetermined quantum gates. Here, we implement the CV-QAOA by developing an automated collaborative computing system between such a programmable CV photonic quantum computer and a classical computer, where the latter assesses the output of the former and automatically updates the program of the former in real time.

A. Concept of our implementation

Our implementation can be conceptually shown in Fig. 1(a). The input state is a p -squeezed state, which approximates $|p = 0\rangle$. After the parametrized operation of the cost and mixer on the input state, the output state is measured in the $\hat{\mathbf{x}}$ basis. According to Eq. (2), $\hat{U}(\boldsymbol{\eta}, \boldsymbol{\gamma})$ in our specific problem setting transforms $\hat{\mathbf{x}}$ as

$$\hat{x}_{\text{out}} = (1 - 2\eta\gamma)\hat{x}_{\text{in}} + \gamma\hat{p}_{\text{in}} + 2a\eta\gamma, \quad (3)$$

($\hat{x}_{\text{in}}, \hat{p}_{\text{in}}$) and ($\hat{x}_{\text{out}}, \hat{p}_{\text{out}}$) being the quadrature amplitude of the input and output states, respectively. The parameters η and γ are iteratively updated according to the sampling results of the output state.

To realize the concept of Fig. 1(a), we design and implement an optical and electrical setup shown in Fig. 1(b). To perform a gate operation of Eq. (3) in a programmable way, we use a measurement-induced gate composed of an ancillary state, measurement, and feedforward. In fact, this setup can be regarded as the simplest version of a more general setup to implement the CV-QAOA for arbitrary-order functions [22], as we show in Appendix B. The following is a specific description of the current setup. The input state is produced by an optical parametric oscillator (OPO) named OPO-1. The wave function of the state is squeezed in the p direction in the sense that $\langle \hat{x}_{\text{in}}^2 \rangle = e^{2r_+}/2$ and $\langle \hat{p}_{\text{in}}^2 \rangle = e^{-2r_-}/2$ with $r_+, r_- > 0$. The ancillary state from OPO-2 is an orthogonally squeezed state with $\langle \hat{x}_a^2 \rangle = e^{-2r_-}/2$ and $\langle \hat{p}_a^2 \rangle = e^{2r_+}/2$. These two fields interfere at the beam splitter having variable transmissivity T and then they are sent to two homodyne detectors. The homodyne detectors measure the quadrature amplitudes $\hat{x}_{1,\theta} = \hat{x}_1 \cos \theta + \hat{p}_1 \sin \theta$ and $\hat{x}_{2,\phi} = \hat{x}_2 \cos \phi + \hat{p}_2 \sin \phi$. Here (\hat{x}_1, \hat{p}_1) and (\hat{x}_2, \hat{p}_2) are the quadrature amplitudes of the upper and lower beams coming out of the variable beam splitter, respectively. The parameters of the processor T , θ , and ϕ can be varied via the applied voltage to the corresponding electro-optic modulators (EOMs). Then $\hat{x}_{1,\theta}$ is fed forward to $\hat{x}_{2,\phi}$ with a certain gain. To let the above-described circuit act as the gate designated by $(\boldsymbol{\eta}, \boldsymbol{\gamma})$ as Eq. (3), the parameters of the optical circuit are determined by $T = 1/(1 + \gamma^2)$, $\tan \theta = \gamma^2/[\gamma - 2\eta(1 + \gamma^2)]$, and $\tan \phi = \gamma$, and correspondingly the feedforward gain is set by $g = \sqrt{\gamma^2 - 4\eta\gamma + 4\eta^2(1 + \gamma^2)}$, in which settings the contribution of the antisqueezed quadrature of the ancilla \hat{p}_a to the gate output is canceled so that the measurement-induced gate operates properly. The constant displacement of $x_d = 2a\eta\gamma$ is also performed for the constant term in Eq. (3). The feedforward and constant displacing are numerically done by postprocessing in the classical computer, which is justified because it gives the same results as those given by optical displacing. In fact, feedforward operations were performed by

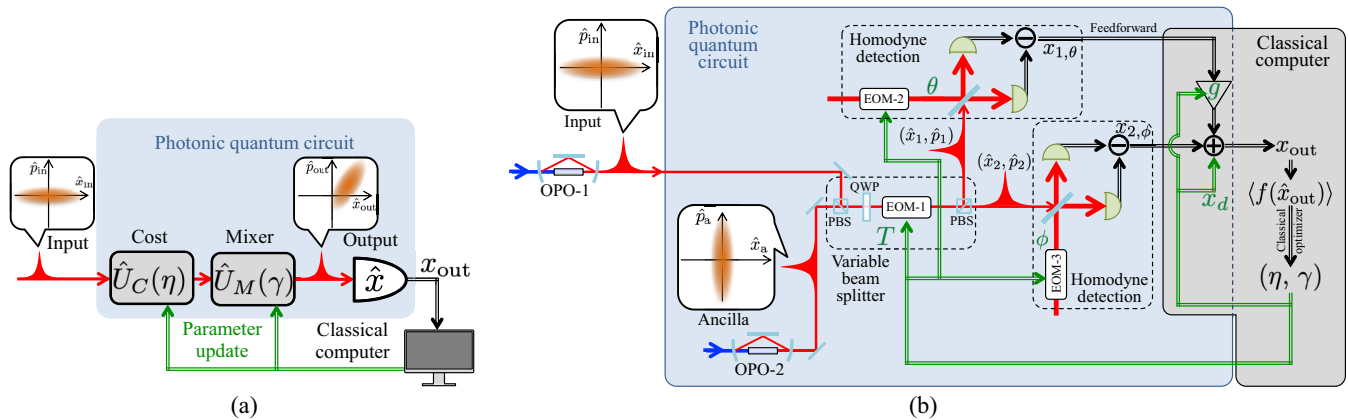


FIG. 1. Experimental implementation of the CV-QAOA. (a) Conceptual diagram of our demonstration. The input state is a squeezed state. The cost operator $\hat{U}_C(\eta)$ and the mixer operator $\hat{U}_M(\gamma)$ are applied to it, and the output state is measured in the \hat{x} basis. The cost and mixer operators are updated following a certain protocol according to the circuit output x_{out} . (b) Experimental setup. The input state is produced by OPO-1, and the ancillary state for the measurement-induced operation is produced by OPO-2. They interfere at the beam splitter having a variable transmissivity of T , and each of the two beams from the beam splitter is measured by the homodyne detector with a programmable measurement basis $\hat{x}_{1,\theta}$ and $\hat{x}_{2,\phi}$. The feedforward operation proportional to $\hat{x}_{1,\theta}$ and the constant displacement operation are applied to $\hat{x}_{2,\phi}$ in a post process, which yields the circuit output x_{out} . The circuit parameters (T, θ, ϕ), the feedforward gain g , and the constant displacement x_d are determined according to a set of (η, γ) so that the measurement-induced gate is performed in a consistent way. EOM, electro-optic modulator; PBS, polarizing beam splitter; QWP, quarter-wave plate.

postprocessing in recent demonstrations of one-way quantum computation [30,31]. After the above classical postprocessing, the gate output \hat{x}_{out} becomes

$$\hat{x}_{\text{out}} = (1 - 2\eta\gamma)\hat{x}_{\text{in}} + \gamma\hat{p}_{\text{in}} + 2\eta\gamma - 2\eta\hat{x}_a, \quad (4)$$

which asymptotically coincides with Eq. (3) in the high squeezing limit $r_- \rightarrow \infty$ in the sense that the variance of the noise term $-2\eta\hat{x}_a$ approaches zero (see Appendix A for the derivations). Therefore, our photonic processor depicted in Fig. 1(b) is capable of obtaining the x -measurement result of the output state, \hat{x}_{out} , with the gate parameters η and γ varied.

B. Experimental setup

Here we describe the details of our experimental setup in Fig. 1(b). We use a continuous-wave laser of wavelength 1545.3 nm. Two OPOs are pumped by the second harmonic fields with wavelength of 772.7 nm. The pump power is set to 200 mW. The full width at half maximum of the OPOs is 60 MHz. The variable beam splitter is composed of a bulk electro-optic modulator named EOM-1, a quarter-wave plate, and a pair of polarizing beam splitters. EOM-1 serves as a variable polarization rotator and thus works as a variable beam splitter with polarization optics. We inserted the quarter-wave plate so that the transmissivity is 50% when no voltage is applied to EOM-1, which makes it easy to lock the relative phase between the input and the ancillary beams.

Each homodyne detection is performed by interfering the local oscillator field with the signal field at a 50:50 beam splitter. Two beams from the beam splitter are received by two photodiodes, the photocurrents of which are subtracted with each other and amplified in the electric circuit. The bandwidth of the circuit is about 200 MHz. The optical power of the local oscillator field is set to 10 mW. A fiber-coupled electro-optic

modulator EOM-2 or 3 shifts the optical phase of each local oscillator for the control of the homodyne angle θ or ϕ .

The outcome of the homodyne detection is acquired by an oscilloscope and then sent to the classical computer. The time series from the oscilloscope is converted to a quadrature amplitude by convoluting it with a mode function $h(t)$ defined by

$$h(t) = \begin{cases} te^{-\Gamma t^2} & (|t| < t_1) \\ 0 & (\text{otherwise}), \end{cases} \quad (5)$$

where $\Gamma = 3 \times 10^7/\text{s}$ and $t_1 = 50$ ns. The purpose of using this mode function is to eliminate undesirable effects from low-frequency electrical noise from homodyne detectors [33]. Based on the measured quadratures and the subsequent analysis, the classical computer can automatically reprogram the photonic quantum circuit by changing the voltages applied to EOM-1, 2, and 3. This enables the classical computer to collaborate with the photonic quantum computer updated in real time and perform the CV-QAOA.

As a preliminary measurement, the outputs of OPO-1 and OPO-2 are measured by the homodyne detectors with the transmissivity of the variable beam splitter set to zero. The squeezing level and the antisqueezing level of these modes are measured to be -5.3 dB and $+9.0$ dB on average. This measurement result indicates that the overall optical loss of the experimental setup is estimated to be 22%.

IV. OPTIMIZATION LANDSCAPE AND ALGORITHM PERFORMANCE

The CV-QAOA is performed on the processor as follows: first, we repeatedly run the circuit and sample \hat{x}_{out} with the parameters η and γ fixed to calculate the mean value $\langle f(\hat{x}_{\text{out}}) \rangle$; then, an outer-loop optimizer in the classical computer suggests new parameters to decrease $\langle f(\hat{x}_{\text{out}}) \rangle$. These two stages

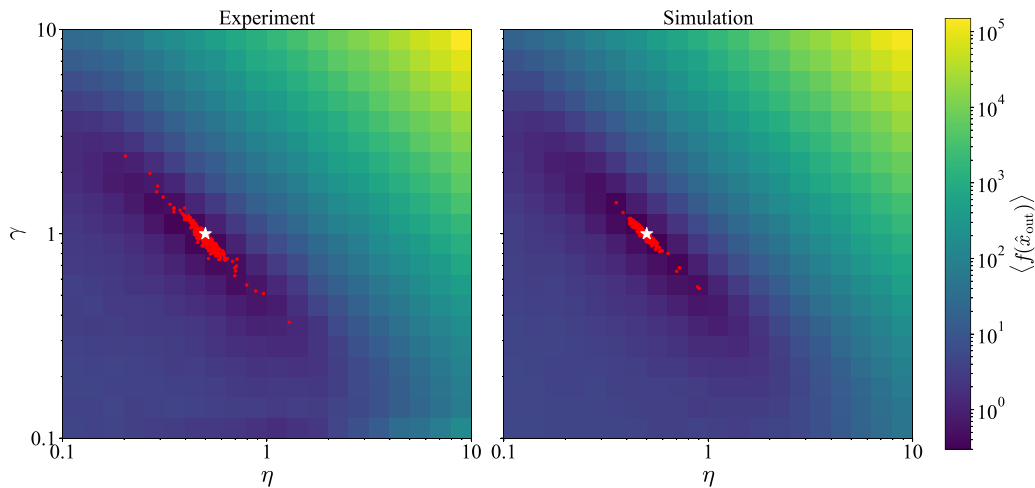


FIG. 2. Experimental and simulated landscapes of the CV-QAOA. The landscape structure of the experiment (left) agrees with that of the simulation (right). Each grid is evaluated by 1000 samples of \hat{x}_{out} . $f(x) = (x - 1)^2$ is used for the landscape evaluations. Each red dot on the landscapes shows the parameter set that records the smallest $\langle f(\hat{x}_{\text{out}}) \rangle$ during the 100-step Bayesian optimization. The white stars denote the theoretical optimum of $(\eta, \gamma) = (1/2, 1)$. See Appendix E for the simulation condition.

are repeated alternately. To experimentally demonstrate the capability of the above, we operate the whole system in three different conditions for the parameter update: (i) the parameters η and γ are scanned like a grid search; (ii) the parameters are fixed at their optimum; and (iii) the parameters are updated according to the protocol of the Bayesian optimization. Each result is shown in the following.

First, to qualitatively diagnose that the processor operates properly for the full search range of the outer-loop optimizer, $\langle f(\hat{x}_{\text{out}}) \rangle$ is evaluated with the parameters η and γ scanned like a grid search. The landscapes of $\langle f(\hat{x}_{\text{out}}) \rangle$ as a function of η and γ are obtained from the experiment or the numerical simulation as shown in Fig. 2. The landscape structure of the experiment reasonably well agrees with that of the simulation. Specifically, $\langle f(\hat{x}_{\text{out}}) \rangle$ is small around $\eta\gamma = 1/2$ and the smallest around $(\eta, \gamma) = (1/2, 1)$, which is the theoretical optimum in the high squeezing limit indicated by white stars in the figure. Note that the optimal point in the finite-squeezing case is $(\eta, \gamma) = (\sqrt{1 - \delta}/2, \sqrt{1 - \delta})$, where $\delta = e^{-2r_-}/(e^{2r_+} + 2a^2)$, and thus the deviation from $(1/2, 1)$ is less than 2% with the values of r_+ and r_- for our experiment no matter the value of a . The overall landscape structure is also independent of the value of a because it only affects the steepness of the valley of $\eta\gamma \sim 1/2$.

Next, to visualize how the algorithm works like the gradient descent method to reshape the wave function of \hat{x} , we run the processor with the parameters fixed at the optimal point $(\eta, \gamma) = (1/2, 1)$. Figure 3 shows the histogram of the sampled \hat{x}_{out} for a specific a . The sampling results of the input state \hat{x}_{in} are also plotted. It can be found that the input state has a broad distribution, and the optimal gate operation localizes the distribution around a , the exact solution. In fact, the standard deviation of \hat{x}_{out} is $0.5767(4)$, smaller than that of the vacuum state $1/\sqrt{2}$ thanks to the quantumness of the processor. Also, the gap between the mean output and the exact solution is $\langle \hat{x}_{\text{out}} \rangle - a = -7.1(5) \times 10^{-3}$, the absolute value of which is much smaller than the standard deviation. This indicates that the systematic shift from the exact solution

is negligible with respect to the statistical broadening of the distribution. These results visually prove that the experimentally implemented gate for $\hat{U}(\eta, \gamma)$ in our problem setting properly localizes the distribution around the minimum like the gradient descent method when the parameters are optimal.

Finally, to evaluate the performance of the CV-QAOA in a realistic condition where the optimum of the parameters is unknown, we perform the algorithm with the parameter updated by the Bayesian optimization. The optimizer suggests new parameters every 1000 samples of \hat{x}_{out} so that $\ln \langle f(\hat{x}_{\text{out}}) \rangle$ is minimized (see Appendix C). We repeat such a process many times with the value of a changed to see the statistical

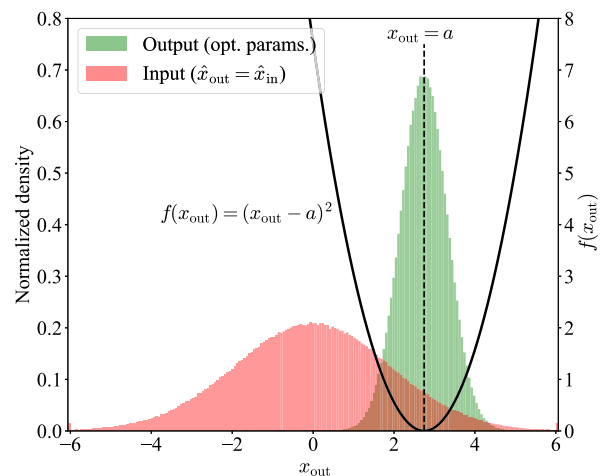


FIG. 3. Histogram of the output distribution. The normalized frequencies of the sampled x_{out} in the experiment are shown. The green bars are the output distribution with the optimal parameters of $(\eta, \gamma) = (1/2, 1)$ while the red ones are the distribution of the input state, which corresponds to $(\eta, \gamma) = (0, 0)$. The number of the sampled x_{out} is 1.1×10^6 for each. The curve of $f(x_{\text{out}}) = (x_{\text{out}} - a)^2$ is overlaid. We use $a = 2.745$ for these measurements. The comparison of two distributions demonstrates how the CV-QAOA works.

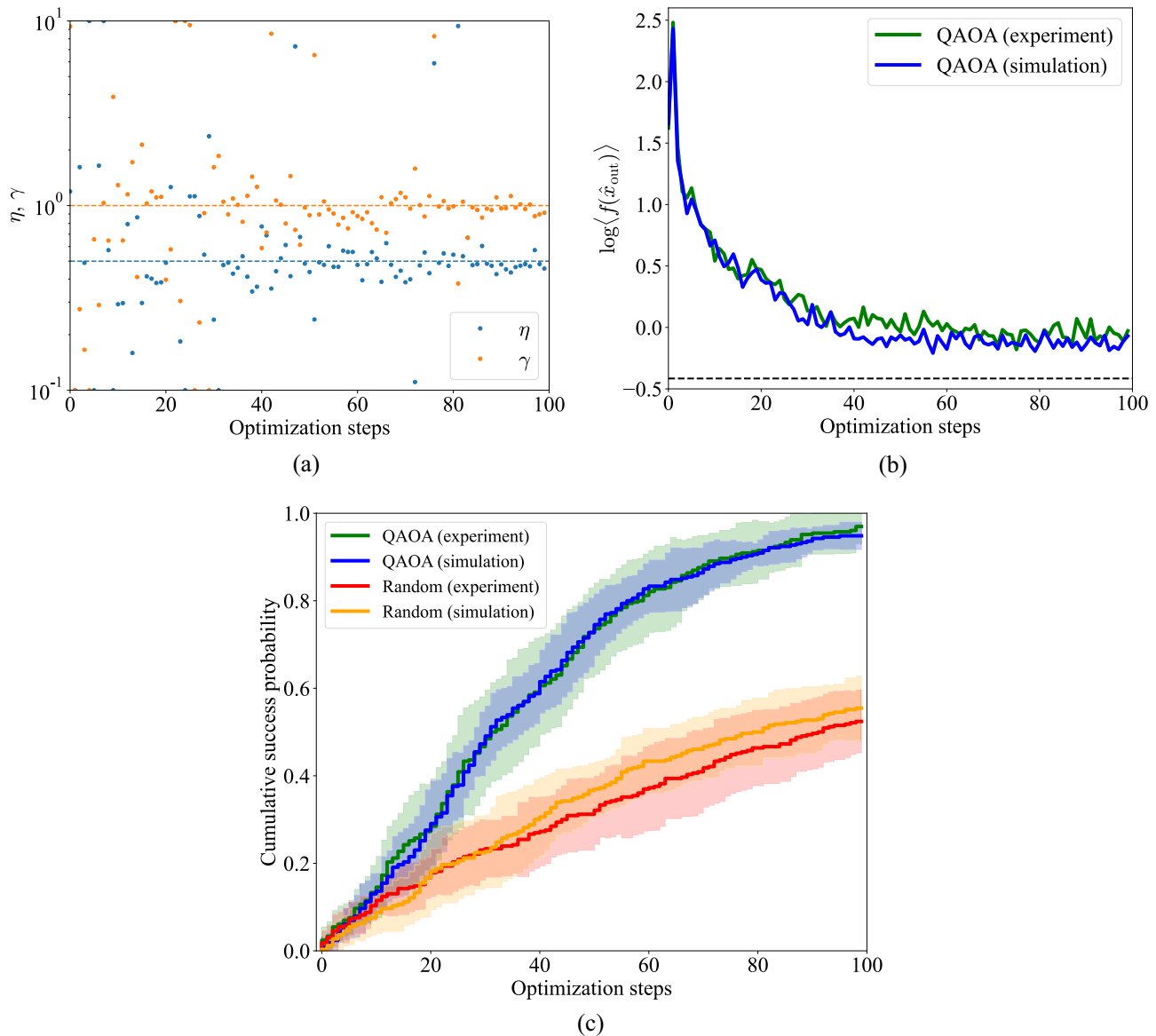


FIG. 4. Algorithm performance with parameter updates. (a) Typical trace of the parameters updated by the Bayesian optimization. The initial search uniformly spans the full range but gradually the search becomes concentrated around the optimum, which is depicted by the dashed lines. (b) Convergence of the classical optimization. The solid traces show the average of the target function $\ln \langle f(\hat{x}_{\text{out}}) \rangle$ from the experiment and the simulation. The black dashed line is the theoretical values for the case where the parameters are fixed at $(\eta, \gamma) = (1/2, 1)$. (c) Success probability to sample x_{out} such that $f(x_{\text{out}}) < 1 \times 10^{-9}$. The solid line is derived by averaging the probability for the 11 sets of such trials. The shaded area shows the $\pm 1\sigma$ region around the average. The green and blue plots are the experimental and simulated results of the CV-QAOA, respectively. As a reference, the red and orange plots show the experimental and simulated results of random sampling. They indicate that the CV-QAOA finds the minimum of $f(x)$ significantly more efficiently than the random sampling.

behavior of the algorithm (see Appendix D). In Fig. 2, the overlaid red dots show the distribution of the classically optimized pair of (η, γ) . Each dot corresponds to the pair of parameters that gives the smallest $\langle f(\hat{x}_{\text{out}}) \rangle$ among 100 suggestions by the optimizer for each execution of the CV-QAOA. These figures show that the classical optimizer reaches the point around the optimum within the 100 steps as the simulation predicts. Figure 4(a) shows the typical trace of the parameter update by the Bayesian optimization, where a wide area is initially explored but gradually the search becomes concentrated around the optimum. The behavior of $\ln \langle f(\hat{x}_{\text{out}}) \rangle$, which is the target function of the Bayesian

optimization, is shown in Fig. 4(b). The traces denote the average of $\ln \langle f(\hat{x}_{\text{out}}) \rangle$ for all the CV-QAOA trials. The observed decreasing trend of the average of the target function is comparable to the simulation expectation. The small differences between the experimental and simulated traces can be attributed to imperfections of the optical setup such as intensity fluctuations and/or alignment drifts of the local oscillator beams for the homodyne detectors.

To quantify the performance in minimizing the function, the success probability of finding the minimum is evaluated experimentally and numerically. Figure 4(c) shows the cumulative success probability of finding the minimum of

$f(x)$ up to a certain step of the parameter updates. Here we set the criterion of the success by $f(x_{\text{out}}) = (x_{\text{out}} - a)^2 < 1 \times 10^{-9}$. The success probability is calculated by the success frequency for 30 different values of a in $f(x) = (x - a)^2$. The solid line is derived by averaging the success probability for 11 sets of the trials while the shaded area denotes the $\pm 1\sigma$ region around the average derived from the 11 trials. These results show that the experimental results of the increase of the cumulative success probability coincide with the numerically simulated ones. They also show that the success probability obtained by the CV-QAOA is significantly better than that by random sampling, where the input state is directly measured.

V. DISCUSSION

In conclusion, we demonstrate the successful implementation of the CV-QAOA with the parameters of the programmable photonic quantum processor updated. The demonstration experimentally shows that the performance of the algorithm for the minimization of quadratic functions is significantly better than that of the random sampling and comparable to what the numerical simulation predicts. Even though the experimental system is influenced by various imperfections including optical loss, the CV-QAOA still successfully finds the minimum of the function, indicating that the algorithm works robustly.

Such imperfections limit the effective squeezing levels. The optimum effective squeezing level for the overall performance is nontrivial. In fact, with higher squeezing levels, the width of the output distribution in Fig. 3 gets narrower and this increases the probability of finding the optimal point of x . On the other hand, it is also observed in our numerical simulation that the parameters η and γ are optimized more slowly with higher squeezing levels due to the landscape change in Fig. 2. The former (the latter) is advantageous (disadvantageous) to the algorithm. Optimum squeezing level should be investigated, but its detailed analysis is left for future work.

Our experiment in this paper can be simulated efficiently with classical computers because our system is entirely built with Gaussian building blocks [34]. However, we show that our implementation can be extended to arbitrary-order functions by adding non-Gaussian ancillary states other than squeezed states [22] (Appendix B). This may push the CV-QAOA to the non-Gaussian regime beyond efficient classical simulation. It can also be extended to multivariable functions by using beam splitters for multimode interactions. In this way, more complex functions can be minimized to address the practical problems. For such more complex functions, we may set $P > 1$ with a deeper circuit at the cost of the increased number of the classical parameters to be optimized (η, γ). The current system took ~ 100 s to perform one CV-QAOA trial with 100 optimization steps. This runtime was mainly dominated by the measurement time, which can be shortened by increasing the bandwidths of the OPOs and the homodyne detectors.

This work is an experimental realization of a quantum algorithm using CV information, which proves the usefulness of CV quantum systems for natively solving CV problems. In general, CV problems can also be solved with qubit-based quantum computers [35], but it requires more resources to

represent discretized CV parameters with many qubits and also introduces quantization errors. CV quantum computers can avoid such difficulties and efficiently encode CV problems. This demonstration sheds light on the advantages of CV quantum computing that its infinite-dimensional space can be exploited in NISQ applications. It stimulates the realizations of other VQAs in CV systems such as quantum machine learning [12,13] and thus opens a promising way toward quantum advantage.

ACKNOWLEDGMENTS

The authors thank Keisuke Fujii, Kosuke Fukui, and Kosuke Mitarai for valuable discussions. This work was partly supported by JSPS KAKENHI Grants No. 20H01833 and No. 21K18593, MEXT Leading Initiative for Excellent Young Researchers, Toray Science Foundation (19-6006), and the Canon Foundation.

APPENDIX A: INPUT-OUTPUT RELATION OF THE OPTICAL CIRCUIT

Let us describe why the optical circuit in Fig. 1(b) works as the unitary operation $\hat{U}(\eta, \gamma) = e^{-i\gamma\hat{p}^2/2} e^{-i\eta(\hat{x}-a)^2}$ for our problem setting and the successive measurement in \hat{x} basis. Here we recall that this unitary transforms the input state represented by $(\hat{x}_{\text{in}}, \hat{p}_{\text{in}})$ into the output state represented by $(\hat{x}_{\text{out}}, \hat{p}_{\text{out}})$ as

$$\begin{bmatrix} \hat{x}_{\text{out}} \\ \hat{p}_{\text{out}} \end{bmatrix} = \begin{bmatrix} 1 - 2\eta\gamma & \gamma \\ -2\eta & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{\text{in}} \\ \hat{p}_{\text{in}} \end{bmatrix} + \begin{bmatrix} 2a\eta\gamma \\ 2a\eta \end{bmatrix}. \quad (\text{A1})$$

One of the possible implementations of such a linear transformation is shown in Fig. 5, which is based on the squeezing gate in Ref. [36]. First, we explain how this measurement-induced implementation works. The ancillary state represented by (\hat{x}_a, \hat{p}_a) is an x -squeezed state. The output of this circuit is calculated as

$$\begin{bmatrix} \hat{x}_{\text{out}} \\ \hat{p}_{\text{out}} \end{bmatrix} = \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \hat{x}_2 \\ \hat{p}_2 + g'\hat{x}_{1,\theta} + x'_d \end{bmatrix} \quad (\text{A2})$$

$$= \begin{bmatrix} \hat{x}_{2,\phi} + g'\hat{x}_{1,\theta} \sin \phi + x'_d \sin \phi \\ \hat{p}_{2,\phi} + g'\hat{x}_{1,\theta} \cos \phi + x'_d \cos \phi \end{bmatrix}, \quad (\text{A3})$$

where g' is the feedforward gain and x'_d is the constant displacement. Here, (\hat{x}_i, \hat{p}_i) and $(\hat{x}_{i,\psi}, \hat{p}_{i,\psi})$ ($i = 1, 2$, and $\psi \in \mathbb{R}$) denote the quadrature amplitudes of the beams coming out of the beam splitter defined by

$$\hat{x}_1 = \sqrt{1-T}\hat{x}_{\text{in}} + \sqrt{T}\hat{x}_a, \quad (\text{A4})$$

$$\hat{p}_1 = \sqrt{1-T}\hat{p}_{\text{in}} + \sqrt{T}\hat{p}_a, \quad (\text{A5})$$

$$\hat{x}_2 = \sqrt{T}\hat{x}_{\text{in}} - \sqrt{1-T}\hat{x}_a, \quad (\text{A6})$$

$$\hat{p}_2 = \sqrt{T}\hat{p}_{\text{in}} - \sqrt{1-T}\hat{p}_a, \quad (\text{A7})$$

$$\begin{bmatrix} \hat{x}_{i,\psi} \\ \hat{p}_{i,\psi} \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{bmatrix} \begin{bmatrix} \hat{x}_i \\ \hat{p}_i \end{bmatrix}. \quad (\text{A8})$$

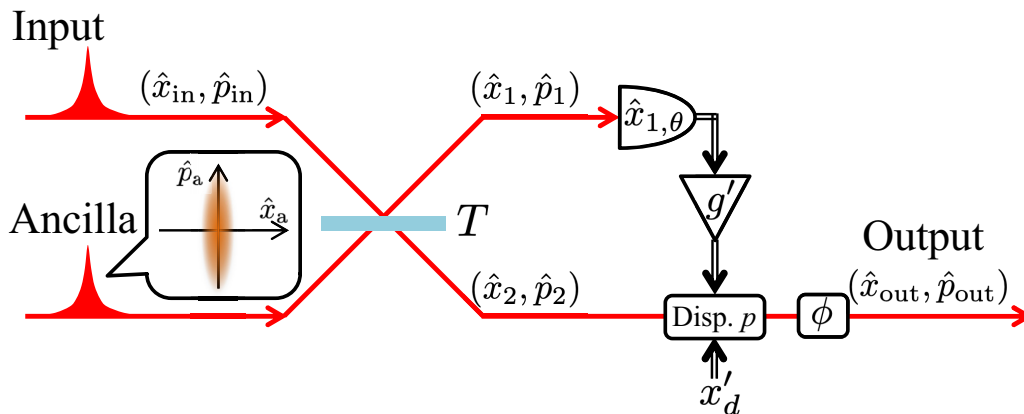


FIG. 5. Optical circuit for a measurement-induced linear transformation. The input state interferes with the ancilla at the beam splitter having the transmissivity of T . One outgoing beam from the beam splitter is measured by homodyne detection, and the measurement outcome is fed forward to the other beam. The quadrature \hat{p}_2 is displaced by $g'\hat{x}_{1,\theta} + x'_d$, and then a phase rotation by ϕ is applied.

If we set the feedforward gain by $g' = \sqrt{(1-T)/T} / \sin \theta$, the output described by Eq. (A3) becomes

$$\begin{aligned} \begin{bmatrix} \hat{x}_{\text{out}} \\ \hat{p}_{\text{out}} \end{bmatrix} &= \frac{1}{\sqrt{T}} \begin{bmatrix} (1-T) \cot \theta \sin \phi + T \cos \phi & \sin \phi \\ (1-T) \cot \theta \cos \phi - T \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \hat{x}_{\text{in}} \\ \hat{p}_{\text{in}} \end{bmatrix} \\ &+ \begin{bmatrix} \sin \phi \\ \cos \phi \end{bmatrix} x'_d + \sqrt{1-T} \begin{bmatrix} \cot \theta \sin \phi - \cos \phi \\ \cot \theta \cos \phi + \sin \phi \end{bmatrix} \hat{x}_a. \end{aligned} \quad (\text{A9})$$

Note that g' is chosen so that the antisqueezed quadrature \hat{p}_a disappears in this expression. This expression indicates that, apart from a noise term proportional to the squeezed quadrature \hat{x}_a , the circuit in Fig. 5 is capable of performing various linear transformations by changing T , θ , ϕ , and x'_d . In fact, by setting $T = 1/(1 + \gamma^2)$, $\tan \theta = \gamma^2/[\gamma - 2\eta(1 + \gamma^2)]$ ($0 \leq \theta < \pi$), $\tan \phi = \gamma$ ($0 \leq \phi < \pi/2$), and $x'_d = 2a\eta\sqrt{1 + \gamma^2}$, Eq. (A9) becomes

$$\begin{aligned} \begin{bmatrix} \hat{x}_{\text{out}} \\ \hat{p}_{\text{out}} \end{bmatrix} &= \begin{bmatrix} 1 - 2\eta\gamma & \gamma \\ -2\eta & 1 \end{bmatrix} \begin{bmatrix} \hat{x}_{\text{in}} \\ \hat{p}_{\text{in}} \end{bmatrix} + \begin{bmatrix} 2a\eta\gamma \\ 2a\eta \end{bmatrix} \\ &+ \begin{bmatrix} -2\eta \\ (\gamma - 2\eta)/\gamma \end{bmatrix} \hat{x}_a, \end{aligned} \quad (\text{A10})$$

which asymptotically coincides with Eq. (A1) in the high squeezing limit of $\hat{x}_a \rightarrow 0$. Note that the upper row of this equation is identical to Eq. (4).

Given that we eventually measure the quadrature amplitude \hat{x}_{out} , the phase rotation by ϕ can be achieved instead by changing the homodyne angle for the measurement of the output state. In addition, the displacing operation can be replaced by numerical postprocessing after the homodyne measurement as the displacing operation only shifts the mean value of the measurement outcome. For this reason, the circuit in Fig. 1(b) is equivalent to the one in Fig. 5 as long as the quadrature amplitude of the output state is measured. In the circuit in Fig. 1(b), \hat{x}_{out} is in fact provided using the two homodyne-measurement outcomes by $\hat{x}_{2,\phi} + g'\hat{x}_{1,\theta} + x_d$, which corresponds to the upper row of Eq. (A3). Here we

redefined $g := g' \sin \phi = \sqrt{\gamma^2 - 4\eta\gamma + 4\eta^2(1 + \gamma^2)}$ and $x_d := x'_d \sin \phi = 2a\eta\gamma$.

APPENDIX B: IMPLEMENTATION OF HIGHER-ORDER FUNCTIONS

Let us show that our setup can be extended to a quantum circuit for minimizing a higher-order function. As can be seen from Eq. (2), if the minimized function $f(x)$ is an n th-order polynomial, the transformation by a pair of the cost and mixer Hamiltonians in the QAOA involves an $(n - 1)$ th-order polynomial of \hat{x} :

$$\hat{x} \rightarrow \hat{x} + \gamma \hat{p} - \eta\gamma \sum_{k=1}^n k a_k \hat{x}^{k-1}, \quad (\text{B1})$$

$$\hat{p} \rightarrow \hat{p} - \eta \sum_{k=1}^n k a_k \hat{x}^{k-1}, \quad (\text{B2})$$

where we express $f(x) = \sum_{k=0}^n a_k x^k$ with real coefficients a_k . The above transformation can be implemented by the circuit shown in Fig. 6. In contrast to Fig. 5, where one beam from the beam splitter is measured by simple homodyne detection, the beam is nonlinearly measured with the help of the ancillary states. The working principle can be made clear by dividing the circuit into two parts: the nonlinear measurement followed by the feedforward, and the phase rotation by ϕ .

The combination of the nonlinear measurement and feedforward provides an $(n - 1)$ th-order polynomial of \hat{x} in the following way. The ancillary states A_k ($3 \leq k \leq n$) are k th-order phase states with reduced fluctuations in quadratures $\hat{p}_k - k\chi_k \hat{x}_k^{k-1}$, where (\hat{x}_k, \hat{p}_k) are canonical pairs of the quadrature operators of the corresponding modes. The beam is sequentially coupled with the ancillary states A_k by the beam splitters having transmissivity of T_k . Then, one outgoing mode from each beam splitter is measured by homodyne detection in \hat{x} quadrature, the outcome of which is labeled as q_k . The transmissivity T_k is adaptively controlled depending on the past outcomes q_j , where $k < j \leq n$. The rightmost homodyne measurement is done with the homodyne angle of θ , which is also controlled depending on the past outcomes. The

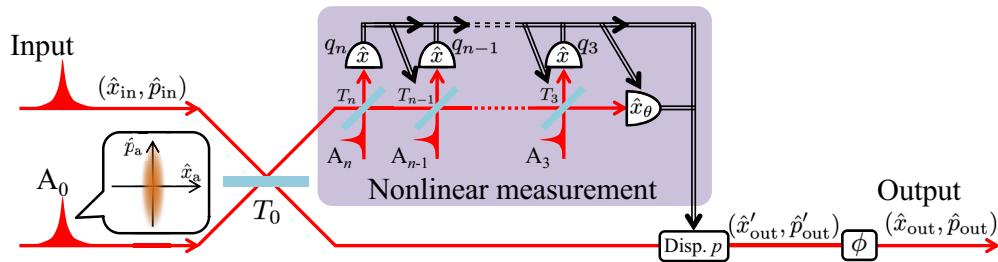


FIG. 6. Implementation of a higher-order function. The outgoing beam from the first beam splitter is nonlinearly measured with the help of the ancillary nonlinear phase states, and the measurement outcomes are fed forward before the phase rotation by ϕ . Instead of also preparing a quadratic phase state and an additional beam splitter at the right end of the nonlinear measurement block as proposed in Fig. 1(d) in Ref. [22], the homodyne angle θ of the rightmost homodyne measurement is made variable. In fact, the implementation of the fourth-order nonlinear phase gate is proposed in this way (Fig. 2 in Ref. [22]).

quadratures after appropriate feedforwards from the nonlinear measurement block are expressed, in the limit that the ancillae are ideal, as

$$\hat{x}'_{\text{out}} = \sqrt{T_0} \hat{x}_{\text{in}}, \quad (\text{B3})$$

$$\hat{p}'_{\text{out}} = \frac{1}{\sqrt{T_0}} \hat{p}_{\text{in}} + \sum_{k=1}^n k C_k \hat{x}_{\text{in}}^{k-1}, \quad (\text{B4})$$

where C_k depends on q_j ($k < j \leq n$), T_l ($k \leq l \leq n$), and θ [see Eqs. (5) and (9) in Ref. [22]]. Here, by recurrently determining T_n, T_{n-1}, \dots, T_3 in the order from n to 3 and then θ , the coefficient C_k ($2 \leq k \leq n$) can be arbitrarily chosen. As the coefficient C_1 corresponds to constant displacing for \hat{p} , it is also arbitrarily determined at the displacing operation. Therefore, the above process realizes n th-order nonlinearity with arbitrary coefficients C_k .

The phase rotation by ϕ transforms $(\hat{x}'_{\text{out}}, \hat{p}'_{\text{out}})$ as

$$\hat{x}_{\text{out}} = \sqrt{T_0} \hat{x}_{\text{in}} \cos \phi + \frac{\hat{p}_{\text{in}}}{\sqrt{T_0}} \sin \phi + \sin \phi \sum_{k=1}^n k C_k \hat{x}_{\text{in}}^{k-1}, \quad (\text{B5})$$

$$\hat{p}_{\text{out}} = -\sqrt{T_0} \hat{x}_{\text{in}} \sin \phi + \frac{\hat{p}_{\text{in}}}{\sqrt{T_0}} \cos \phi + \cos \phi \sum_{k=1}^n k C_k \hat{x}_{\text{in}}^{k-1}. \quad (\text{B6})$$

Here, by setting $\tan \phi = \gamma$, $T_0 = 1/(1 + \gamma^2)$, $C_2 = [\gamma/2 - a_2 \eta(1 + \gamma^2)]/\sqrt{1 + \gamma^2}$, and $C_k = -a_k \eta \sqrt{1 + \gamma^2}$ ($k \neq 2$), these equations become

$$\hat{x}_{\text{out}} = \hat{x}_{\text{in}} + \gamma \hat{p}_{\text{in}} - \eta \gamma \sum_{k=1}^n k a_k \hat{x}_{\text{in}}^{k-1}, \quad (\text{B7})$$

$$\hat{p}_{\text{out}} = \hat{p}_{\text{in}} - \eta \sum_{k=1}^n k a_k \hat{x}_{\text{in}}^{k-1}, \quad (\text{B8})$$

which realize the transformation of Eqs. (B1) and (B2). In this way, the transformation by a pair of the cost and mixer Hamiltonians for a given polynomial $f(x)$ can be implemented. Thus, in the viewpoint of implementing generic higher-order functions, our experimental setup can be regarded as the simplest case of Fig. 6.

APPENDIX C: PARAMETER SEARCH CONDITION WITH THE BAYESIAN OPTIMIZATION

For the update of the circuit parameters (η, γ) in Fig. 1, we adopt the Bayesian optimization, which has been commonly used among the derivative-free optimization methods in the parameter optimization of the QAOA [6,7]. The following conditions of the Bayesian optimization are set after checking the convergence of the circuit parameters in numerical simulations. $\log_{10} \eta$ and $\log_{10} \gamma$ are handed to the Bayesian optimizer as free parameters. The target function to be optimized is $\ln(f(\hat{x}_{\text{out}}))$. We use a PYTHON package for the Bayesian optimization [37]. The Matérn kernel with $\nu = 2.5$ is chosen as the covariance kernel function. The acquisition function is a type called the upper confidence bound given by $\mu + \kappa(t)\sigma$, where μ and σ are the estimated mean and standard deviation of the target function, respectively. We set $\kappa(t) = \kappa_0 \times 0.97^t$, where $\kappa_0 = 2.576$ and t is the number of optimization steps.

As for the QAOA of the qubit system, the parameter range of the unitary operations is often limited by $[0, 2\pi)$. However, the CV case does not have such a limit. Since it is difficult to optimize parameters with unlimited search range, we limit the range by estimating the order of the optimum of the parameters using a generic prescription described below. The search range of the parameters is set by $0.1 < \eta < 10$ and $0.1 < \gamma < 10$ because the optimum is estimated to be $(\eta_{\text{opt}}, \gamma_{\text{opt}}) \sim (1, 1)$.

Here we explain the prescription for the estimation. Let us consider the CV-QAOA with $P = 1$ for a one-variable function $f(x)$. If implemented by a combination of measurement-induced gates [22], a pair of the cost and mixer operations transforms the quadratures as

$$\hat{x}_{\text{out}} = \hat{x}_{\text{in}} + \gamma \hat{p}_{\text{in}} - \eta \gamma \nabla f(\hat{x}_{\text{in}}) + N(\eta, \gamma; \hat{x}_{\text{a}}, \hat{p}_{\text{a}}), \quad (\text{C1})$$

where N is a noise term arising from nonideal ancillary states. \hat{x}_{a} and \hat{p}_{a} formally denote the quadrature amplitudes of the ancillary states. Let us assume that the ancillary states are linearly or nonlinearly squeezed sufficiently and thus $|\langle N \rangle| \ll |x_{\text{min}}|$, where $x_{\text{min}} = \text{argmin} f(x)$. We can also assume that the optimal parameters, namely, $(\eta_{\text{opt}}, \gamma_{\text{opt}})$, localize the distribution of \hat{x}_{out} around x_{min} , and thus provide two conditions: $\langle \hat{x}_{\text{out}} \rangle \sim x_{\text{min}}$ and $\langle \Delta \hat{x}_{\text{out}}^2 \rangle$ is minimized. By assuming that the optimal parameters $(\eta_{\text{opt}}, \gamma_{\text{opt}})$ are adopted and taking the expectation value of Eq. (C1), the assumption $\langle \hat{x}_{\text{out}} \rangle \sim x_{\text{min}}$

reduces to

$$\eta_{\text{opt}}\gamma_{\text{opt}} \sim -\frac{x_{\text{min}}}{\langle \nabla f(\hat{x}_{\text{in}}) \rangle}. \quad (\text{C2})$$

Here we assume $\langle \hat{x}_{\text{in}} \rangle = 0$ and $\langle \hat{p}_{\text{in}} \rangle = 0$ since the input state is the squeezed vacuum. In this way, if the prior information on the order of magnitude of x_{min} and $\langle \nabla f(\hat{x}_{\text{in}}) \rangle$ is available, the product $\eta_{\text{opt}}\gamma_{\text{opt}}$ can be inferred. Let us next consider the variance of \hat{x}_{out} . Since Eq. (C2) estimates the product of the optimal parameters, we evaluate the variance under the constraint of $\eta\gamma = c$, where c denotes the estimated value of the product $\eta_{\text{opt}}\gamma_{\text{opt}}$. By substituting $\eta\gamma = c$ into Eq. (C1) to eliminate η , we have

$$\begin{aligned} \langle \Delta \hat{x}_{\text{out}}^2 \rangle &\sim \langle \Delta [\hat{x}_{\text{in}} - c \nabla f(\hat{x}_{\text{in}})]^2 \rangle \\ &+ \gamma^2 \langle \hat{p}_{\text{in}}^2 \rangle + \left\langle \Delta N^2 \left(\frac{c}{\gamma}, \gamma; \hat{x}_{\text{a}}, \hat{p}_{\text{a}} \right) \right\rangle. \end{aligned} \quad (\text{C3})$$

As the first term does not depend on γ , γ_{opt} can be estimated by minimizing the sum of the second and third terms. The third term can be calculated if the specific form of $N(\eta, \gamma; \hat{x}_{\text{a}}, \hat{p}_{\text{a}})$ is known. Once the third term is calculated, the sum of the second and third terms becomes a one-variable function of γ , which should be able to be minimized.

Let us consider the specific case of $f(x) = (x - a)^2$, and estimate the order of magnitude of the optimal parameters by using the typical magnitude of \hat{x}_{in} . We denote $\sqrt{\langle \hat{x}_{\text{in}}^2 \rangle} = \sigma$, which we regard as the typical magnitude of \hat{x}_{in} . As the function $f(x)$ is quadratic in x and its leading term is x^2 , the order of magnitude of the variation of $f(x)$ in the range of $\pm\sigma$ can be estimated by σ^2 . We then estimate the gradient of the function by dividing the typical range of the function output σ^2 by the typical range of the function input σ : $|\langle \nabla f(x) \rangle| \sim \sigma^2/\sigma = \sigma$. We can also assume that $|x_{\text{min}}| \sim \sigma$. Using these estimations, we can estimate $\eta_{\text{opt}}\gamma_{\text{opt}}$ as

$$\eta_{\text{opt}}\gamma_{\text{opt}} \sim 1 \quad (\text{C4})$$

from Eq. (C2). Let us then calculate the variance under $\eta\gamma = 1$. Since $N(\eta, \gamma; \hat{x}_{\text{a}}, \hat{p}_{\text{a}}) = -2\eta\hat{x}_{\text{a}}$ from Eq. (4), the sum of the second and third terms of Eq. (C3) becomes $\gamma^2 \langle \hat{p}_{\text{in}}^2 \rangle + (4/\gamma^2) \langle \hat{x}_{\text{a}}^2 \rangle$. By minimizing this in terms of γ , γ_{opt} can be estimated as

$$\gamma_{\text{opt}} \sim (4 \langle \hat{x}_{\text{a}}^2 \rangle / \langle \hat{p}_{\text{in}}^2 \rangle)^{1/4} = \sqrt{2} \sim 1. \quad (\text{C5})$$

In this way, the optimal parameters are estimated as $(\eta_{\text{opt}}, \gamma_{\text{opt}}) \sim (1, 1)$, which is why we set the search range by $[0.1, 10]^{\otimes 2}$ with a margin of about a factor of 10.

The above discussion is only a rough estimation of the order of magnitude, and although it works in our case, there

is no guarantee that the optimal parameters exist in the search range. For this reason, a general strategy may be to perform the CV-QAOA within the initial search range and, if there seems to be the optimal point outside the search range, expand or change the search range by observing the behavior of the parameter optimization.

APPENDIX D: REPEATED EXECUTION OF THE CIRCUITS

In the demonstration of the CV-QAOA with parameters updated (Fig. 4), the iterations of the circuit execution are hierarchical. For clarity, the conditions for that hierarchical execution are summarized here. For each pair of the parameters, we repeatedly run the circuit and sample \hat{x}_{out} 1000 times with the parameters fixed to obtain $\langle f(\hat{x}_{\text{out}}) \rangle$. The Bayesian optimizer suggests 100 pairs of the parameters by using the results of $\langle f(\hat{x}_{\text{out}}) \rangle$. Such 100-step classical optimization is repeated 11 times for the same value of a [in $f(x) = (x - a)^2$]. Finally, this repeat is done for 30 different values of a .

The values of a are randomly sampled from the Gaussian distribution of zero mean and the standard deviation equal to 1.99, which corresponds to $\sqrt{\langle \hat{x}_{\text{in}}^2 \rangle}$. This is because we intend to mimic the following situation. Suppose that the range of x that gives the minimum of $f(x)$ (x_{min}) can be roughly estimated by some conditions in the problem settings such as, for example, physical conditions or features of $f(x)$. In this case, the distribution of the initial state can be set so that it covers the estimated range for x_{min} . In this demonstration, mimicking the situation where the range estimation is correct and the solution a is in fact in the estimated range, we repeatedly execute the CV-QAOA with many different a sampled from that range to statistically evaluate the performance of the algorithm. Note that the range for x_{min} can always be matched to the range of the initial state distribution by rescaling and translating x , and resultantly redefining $f(x)$. Generally, if the estimation is incorrect, one can perform the algorithm again with an effectively broader initial state.

APPENDIX E: SIMULATION CONDITION

In the simulation, the optical circuit is numerically simulated by expressing the quadratures of the initial and ancillary states using Gaussian random numbers. Based on the measured squeezing level, we set $\langle \hat{x}_{\text{in}}^2 \rangle = \langle \hat{p}_{\text{a}}^2 \rangle = 10^{9.0/10}/2$ and $\langle \hat{p}_{\text{in}}^2 \rangle = \langle \hat{x}_{\text{a}}^2 \rangle = 10^{-5.3/10}/2$. The asymmetry between the squeezing and antisqueezing implies that optical loss is taken into account. No other imperfections are included in the simulation.

- [1] J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
 [2] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, *Nat. Rev. Phys.* **3**, 625 (2021).

- [3] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, [arXiv:1411.4028](https://arxiv.org/abs/1411.4028).
 [4] 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).

- [5] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum circuit learning, *Phys. Rev. A* **98**, 032309 (2018).
- [6] J. S. Otterbach, R. Manenti, N. Alidoust, A. Bestwick, M. Block, B. Bloom, S. Caldwell, N. Didier, E. S. Fried, S. Hong, P. Karalekas, C. B. Osborn, A. Papageorge, E. C. Peterson, G. Prawiroatmodjo, N. Rubin, C. A. Ryan, D. Scarabelli, M. Scheer, E. A. Sete *et al.*, Unsupervised machine learning on a hybrid quantum computer, [arXiv:1712.05771](https://arxiv.org/abs/1712.05771).
- [7] A. Bengtsson, P. Vikstål, C. Warren, M. Svensson, X. Gu, A. F. Kockum, P. Krantz, C. Križan, D. Shiri, I.-M. Svensson, G. Tancredi, G. Johansson, P. Delsing, G. Ferrini, and J. Bylander, Improved success probability with greater circuit depth for the quantum approximate optimization algorithm, *Phys. Rev. Appl.* **14**, 034010 (2020).
- [8] G. Pagano, A. Bapat, P. Becker, K. S. Collins, A. De, P. W. Hess, H. B. Kaplan, A. Kyprianidis, W. L. Tan, C. Baldwin, L. T. Brady, A. Deshpande, F. Liu, S. Jordan, A. V. Gorshkov, and C. Monroe, Quantum approximate optimization of the long-range Ising model with a trapped-ion quantum simulator, *Proc. Natl. Acad. Sci. USA* **117**, 25396 (2020).
- [9] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo, M. Broughton, B. B. Buckley, D. A. Buell, B. Burkett, N. Bushnell, Y. Chen, Z. Chen, Ben Chiaro, R. Collins, W. Courtney *et al.*, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, *Nat. Phys.* **17**, 332 (2021).
- [10] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, *Nature (London)* **567**, 209 (2019).
- [11] G. Verdon, J. M. Arrazola, K. Brádler, and N. Killoran, A quantum approximate optimization algorithm for continuous problems, [arXiv:1902.00409](https://arxiv.org/abs/1902.00409).
- [12] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd, Continuous-variable quantum neural networks, *Phys. Rev. Res.* **1**, 033063 (2019).
- [13] J. M. Arrazola, T. R. Bromley, J. Isaac, C. R. Myers, K. Brádler, and N. Killoran, Machine learning method for state preparation and gate synthesis on photonic quantum computers, *Quantum Sci. Technol.* **4**, 024004 (2019).
- [14] T. Volkoff, Z. Holmes, and A. Sornborger, Universal compiling and (no-)free-lunch theorems for continuous-variable quantum learning, *PRX Quantum* **2**, 040327 (2021).
- [15] M. Stechly, N. Bashige, and P. Chojacki, Approaching graph problems with continuous variable quantum computing, [arXiv:1906.07047](https://arxiv.org/abs/1906.07047).
- [16] S. Takeda and A. Furusawa, Toward large-scale fault-tolerant universal photonic quantum computing, *APL Photon.* **4**, 060902 (2019).
- [17] C. S. Hamilton, R. Kruse, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex, Gaussian Boson sampling, *Phys. Rev. Lett.* **119**, 170501 (2017).
- [18] H.-S. Zhong, Yu.-H. Deng, J. Qin, H. Wang, M.-C. Chen, Li.-C. Peng, Yi.-H. Luo, D. Wu, Si.-Q. Gong, H. Su, Yi. Hu, P. Hu, X.-Y. Yang, W.-J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You *et al.*, Phase-programmable Gaussian Boson sampling using stimulated squeezed light, *Phys. Rev. Lett.* **127**, 180502 (2021).
- [19] L. S. Madsen, F. Laudenbach, M. F. Askarani, F. Rortais, T. Vincent, J. F. F. Bulmer, F. M. Miatto, L. Neuhaus, L. G. Helt, M. J. Collins, A. E. Lita, T. Gerrits, S. W. Nam, V. D. Vaidya, M. Menotti, I. Dhand, Z. Vernon, N. Quesada, and J. Lavoie, Quantum computational advantage with a programmable photonic processor, *Nature (London)* **606**, 75 (2022).
- [20] J. Huh, G. G. Guerreschi, B. Peropadre, J. R. McClean, and A. Aspuru-Guzik, Boson sampling for molecular vibronic spectra, *Nat. Photon.* **9**, 615 (2015).
- [21] J. M. Arrazola and T. R. Bromley, Using Gaussian Boson sampling to find dense subgraphs, *Phys. Rev. Lett.* **121**, 030503 (2018).
- [22] P. Marek, R. Filip, H. Ogawa, A. Sakaguchi, S. Takeda, J. Yoshikawa, and A. Furusawa, General implementation of arbitrary nonlinear quadrature phase gates, *Phys. Rev. A* **97**, 022329 (2018).
- [23] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, *Phys. Rev. X* **10**, 021067 (2020).
- [24] M. Willsch, D. Willsch, F. Jin, H. De Raedt, and K. Michielsen, Benchmarking the quantum approximate optimization algorithm, *Quant. Inf. Proc.* **19**, 197 (2020).
- [25] G. Cornuéjols, J. Peña, and R. Tütüncü, *Optimization Methods in Finance* (Cambridge University Press, Cambridge, UK, 2006).
- [26] C. C. Aggarwal, *Linear Algebra and Optimization for Machine Learning* (Springer International Publishing, Cham, 2020).
- [27] S. S. Rao, *Engineering Optimization: Theory and Practice* (Wiley, New York, 1996).
- [28] A. K. Pati, S. L. Braunstein, and S. Lloyd, Quantum searching with continuous variables, [arXiv:quant-ph/0002082](https://arxiv.org/abs/quant-ph/0002082).
- [29] K. Fukui and S. Takeda, Building a large-scale quantum computer with continuous-variable optical technologies, *J. Phys. B: At. Mol. Opt. Phys.* **55**, 012001 (2022).
- [30] W. Asavanant, B. Charoensombutamon, S. Yokoyama, T. Ebihara, T. Nakamura, R. N. Alexander, M. Endo, J. Yoshikawa, N. C. Menicucci, H. Yonezawa, and A. Furusawa, Time-domain-multiplexed measurement-based quantum operations with 25-MHz clock frequency, *Phys. Rev. Appl.* **16**, 034005 (2021).
- [31] M. V. Larsen, X. Guo, C. R. Breum, J. S. Neergaard-Nielsen, and U. L. Andersen, Deterministic multi-mode gates on a scalable photonic quantum computing platform, *Nat. Phys.* **17**, 1018 (2021).
- [32] Y. Enomoto, K. Yonezu, Y. Mitsuhashi, K. Takase, and S. Takeda, Programmable and sequential Gaussian gates in a loop-based single-mode photonic quantum processor, *Sci. Adv.* **7**, eabj6624 (2021).
- [33] J. Yoshikawa, S. Yokoyama, T. Kaji, C. Sornphiphatphong, Yu. Shiozawa, K. Makino, and A. Furusawa, Invited Article: Generation of one-million-mode continuous-variable cluster state by unlimited time-domain multiplexing, *APL Photon.* **1**, 060801 (2016).

- [34] 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).
- [35] S. Arai, H. Oshiyama, and H. Nishimori, Quantum annealing for continuous-variable optimization: How is it effective? [arXiv:2305.06631](https://arxiv.org/abs/2305.06631).
- [36] K. Miyata, H. Ogawa, P. Marek, R. Filip, H. Yonezawa, J. Yoshikawa, and A. Furusawa, Experimental realization of a dynamic squeezing gate, *Phys. Rev. A* **90**, 060302(R) (2014).
- [37] F. Nogueira, Bayesian Optimization: Open source constrained global optimization tool for Python, <https://github.com/fmfn/BayesianOptimization>.