Quantum tomography is the procedure that reconstructs the quantum state of a system from repeated measurements of a (complete) set of observables on a number of copies of equally prepared quantum systems. This is the only way to measure a quantum state: indeed, on one hand it is impossible to recover the state from a single copy of the system [1]; on the other, without the measurement of a complete set of observables (a quorum), there is not enough information for the reconstruction as different states may give the exact same statistics on an incomplete set of observables. The same quantum systems may have different possible quorums [2]. For the state of a single mode of the radiation field there are two quorums that are typically used: either the field quadratures \( X_\phi = \frac{1}{2}(a^\dagger e^{i\phi} + ae^{-i\phi}) \) [3–5] or the displaced parity operator \( D(\alpha)(-1)^{a^\dagger a}D(\alpha) \) [6], where \( a \) is the annihilation operator of the mode and \( D(\alpha) = e^{\alpha a^\dagger - \bar{\alpha}a} \) is the displacement operator. In this paper we will focus on the former, which can be detected straightforwardly with a homodyne detector [5]. The homodyne measurements are the marginal probability distributions of each quadrature \( X_\phi \), collectively giving the Radon transform of the Wigner function. As such, the seminal early homodyne tomography method employed Radon transform inversion techniques [7]. This method, thus, requires the homodyne data to be binned in order to obtain a distribution. This introduces a bias in the reconstruction, due to the width of the binning. The Homodyne Computed Tomography (HCT) method, proposed in [8] and refined in [9–11], avoids this problem by directly reconstructing the quantum state—or expectation values of arbitrary operators—without going through the Wigner function. Hence, one could in principle use a single data point for each homodyne measurement (no binning) to ensure complete unbiasedness. Alternative procedures use maximum likelihood methods [12], which produce very high-quality reconstructions, but they are biased and, especially, completely unsuited for the high-dimensional states that we consider here: the minimization procedures entailed become rapidly intractable for large dimensions.

In this paper we provide a (small) further refinement of the HCT method to tame numerical instabilities, and present a Julia package from which some extremely high-dimensional Monte Carlo simulations of reconstructions are obtained to illustrate the method. These show that quantum states with density matrices of one order of magnitude larger than the size of previous simulations and experiments can be reconstructed. This is timely since quantum tomography is starting to be used in microwave cavities [13–16], where even small signals entail huge number of photons, i.e. a high-dimensional subspace of the radiation Hilbert space because of the low energy of each microwave photon.

The outline follows: in Section I we provide a bare-bones description of the method; in Section II we describe how one can produce a numerical algorithm that implements it, and we illustrate it with simulated experiments; in Section III we describe the software packages used for this work; and, finally, in Section IV we draw our conclusions regarding the work presented in this paper. In Appendix A we review the derivation of the method and in Appendix B we derive the optimized recursion relations for the calculation of the Wigner function.

I. HOMODYNE TOMOGRAPHY

The homodyne detector is an apparatus that measures the quadrature observable \( X_\phi \) at optical frequencies. Similar devices can be built for microwave cavities, e.g. [14, 16]. This apparatus receives as input a number \( \varphi \in [0, \pi] \), which typically refers to the phase of a local oscillator in an intense coherent state. It outputs a real number \( x \), which (once calibrated) denotes the measured value of the quadrature \( X_\phi \) identified by \( \varphi \). The data taking stage of the homodyne tomography experiment then resorts in choosing \( n_\varphi \) values \( \varphi_i \) of the phase and measuring \( n_q \) quadrature values \( x_j \) for each chosen phase. Ideally, one should choose \( n_q = 1 \) (i.e. use a different phase for each quadrature measurement) to avoid a possible source of bias, but it is known [17], and confirmed by our simulations (e.g. Fig. 1), that this is not a big concern if the number of phases \( n_\varphi \) is sufficiently large (more excited states requiring a larger number of phases). Using fixed phases simplifies the experiment considerably. The \( n_\varphi \) phases must be chosen uniformly in the interval \( [0, \pi] \) or,
more conveniently for the numerical implementation, in the interval $[0, 2\pi]$; due to the phase space symmetries, the two intervals are equivalent as discussed below.

The result of the experiment is a set of data \{(\varphi_k, x_k)\}_{k \in I_N}$, $I_N = \{0, \ldots, N - 1\}$, where $x_k$ is the result of the $k$-th measurement in which the quadrature $X_{\varphi_k}$ was measured and $N = n_p n_q$ is the total number of measurements.

Now we detail how this data can be used for the tomographic reconstruction. Importantly, the HCT method allows for the direct reconstruction of the expectation value $\langle O \rangle$ of any operator, not necessarily an observable. As an illustrative example, we now show how one can reconstruct the density matrix $\rho_{nm} = \langle n | \rho | m \rangle$ written on the Fock basis. This is achieved by using the non-Hermitian operator $O = |m\rangle \langle n|$, since $\langle O \rangle = \text{Tr}[\rho |m\rangle \langle n|] = \langle n | \rho | m \rangle$. It can be shown (see Appendix, Eq. (A4)) that the matrix element can be written as

$$\rho_{nm} = \frac{1}{\pi} \int_0^\pi d\varphi e^{-i(m-n)\varphi} \int_R dx p_{\varphi}(x)f_{n,m}(x),$$

where $f_{n,m}$ are pattern functions defined below and $p_{\varphi}(x) = \varphi(x)|\varphi\rangle \langle \varphi|$ is the conditional probability of obtaining outcome $x$ given that the $\varphi$-quadrature was measured, namely the outcome relative to the eigenstate $|\varphi\rangle$ of $X_{\varphi}$. Since, by construction, the phases $\varphi_i$ are uniform in the interval $[0, \pi]$, we can interpret $1/\pi$ as the uniform probability of choosing the phase $\varphi$; then, from Bayes rule, the joint probability of choosing a phase $\varphi$ and obtaining a $\varphi$-quadrature outcome $x$ is $p_{\varphi}(x)/\pi$. Thus we can use the collected data $\{(\varphi_k, x_k)\}_{k \in I_N}$ to calculate the double integral of Eq. (1) as a Monte Carlo integral:

$$\rho_{n,m} = \frac{1}{N} \sum_{k=0}^{N-1} e^{-i(m-n)\varphi_k} f_{n,m}(x_k) \pm \Delta_N,$$

where $\Delta_N$ is the error due to the finite $N$. Indeed, the equality is strict only in the limit $N \rightarrow \infty$ (when $\Delta_N \rightarrow 0$), which of course cannot be reached experimentally. However, since we are using Monte Carlo integration, the error $\Delta_N$ for finite $N$ is strictly a statistical error only, which can be easily evaluated from the variance $\sigma^2$ of the data:

$$\Delta_N = \sqrt{\frac{\sigma^2}{N}} = \sqrt{\frac{1}{N(N-1)} \left[ \sum_k F^2(\varphi_k, x_k) - (\sum_k F(\varphi_k, x_k))^2 \right]},$$

with $F(\varphi_k, x_k) = e^{-i(m-n)\varphi_k} f_{n,m}(x_k)$ (more rigorously, if $m \neq n$, when $F$ is complex, one has to separately evaluate the variance of the real and imaginary parts).

## II. NUMERICAL IMPLEMENTATION

The calculation of the pattern functions [11, 18] is quite involved and is reviewed in Appendix A. Starting from the recursion relations derived in the same section, we provide here a simplified form, suitable for numerical evaluation also with non-wide floating point representation, such as the standard 32 bits single precision data type.

To speed up the data analysis, we create a probability distribution by discretizing the possible values of the quadrature $x$ and binning the $n_q$ values of the quadrature data $x_k$ into $n_{\text{bin}}$ bins. As discussed above, in contrast to methods based on the inversion of the Radon transform, this is not a fundamental limitation. Indeed, we show in Fig. 2 that our method still works in the regime where $n_{\text{bin}} \gg n_q$, where most bins are unpopulated and at most a handful of data points end up in the same bin. In this regime, the Radon transform inversion would fail, in contrast to HCT. Nonetheless, binning may be useful, since it allows for a faster and more effective reconstruction because one has to calculate the recurrence relation below only once per bin (this is of interest for the reconstruction of huge data sets).

The binning is done by collecting the data into a $n_{\varphi} \times n_{\text{bin}}$ matrix $S$, called sinogram. Each matrix row $j \in I_{n_{\varphi}}$ represents the quadrature probability at phase $\varphi_j$, namely $S_{j,i}$ is the fraction of the $n_q$ values of $x_k$ that fall into the $i$-th bin, $i \in I_{n_{\text{bin}}}$. Asymptotically we have that $S_{j,i} \sim p_{\varphi_j}(x_i)$, with $x_i$ a representative point in the $i$-th bin. We choose equispaced phases $\varphi_j = \frac{2\pi}{n_{\varphi}} j$. Again, this is a matter of numerical convenience: in principle, one should choose the phases with uniform random probability.
When the samples are taken only in the interval $[0, 2\pi]$, we assume the measurements are taken on the whole interval $[0, 2\pi]$ so that we can use the FFT algorithm; if this is not the case—as it is when the samples are taken only in $[0, \pi]$—one can always double the data exploiting the phase space symmetry $X_{\psi + \pi} = -X_{\psi}$.

In order to implement the tomographic formula on a computer we have to choose a cutoff for the density matrix. As will be clear in the following, this cutoff does not introduce any bias in the reconstruction, although a sensible (i.e. normalized) reconstruction will be obtained only if such cutoff is chosen sufficiently large. Indeed, one can check a posteriori whether the chosen $M$ is sufficient by looking at the normalization of the reconstructed state, namely by checking if it is compatible with one within the statistical error bars. In the following we will assume the density matrix has dimensions $M \times M$.

By dividing the sum into two parts (along the bins and along the phases), the Monte Carlo integral of Eq. (2) becomes

$$\rho_{n,m} = \sum_{i=0}^{n_{\text{bin}}-1} \tilde{S}_{m-n,i} f_{n,m}(x_i),$$

where the factor $1/n_{\text{bin}}$ is included into the definition of $\tilde{S}$ and

$$\tilde{S}_{d,i} = \frac{1}{n_{\varphi}} \sum_{j=0}^{n_{\varphi}-1} S_{j,i} e^{-2\pi i j d}$$

is the one-dimensional discrete Fourier transform (DFT) along the first dimension of $S_{j,i}$.

The pattern functions $f_{n,m}$ can be written in the following factorized form [10, 11, 18] (see Appendix A):

$$f_{n,m}(x) = 2xw_n(x)v_m(x) - \sqrt{n+w_{n+1}(x)v_m(x)} - \sqrt{m+w_n(x)v_{m+1}(x)},$$

where $w_n$ and $v_m$ appear always multiplied together, which means that we can change their definition with respect to their previously published values [11, 18] (see Eqs. (A14), (A16) and (A18)) for the purposes of the numerical implementation. Furthermore, the square roots $\sqrt{\tilde{r}}$ appear always with the same index of either $u_n$ or $v_m$, so we introduce additional vectors $\tilde{u}_n = \sqrt{n}u_n$, $\tilde{v}_m = \sqrt{m}v_m$ that can be precomputed along with $u_n$ and $v_m$. Thus, a recursive definition of $u_n$ that is better suited for numerics is

$$\begin{cases}
  u_0 = \beta, & \tilde{u}_0 = 0, \\
  u_1 = 2xu_0, & \tilde{u}_1 = u_1, \\
  \tilde{u}_n = 2xu_{n-1} - \sqrt{n-2u_{n-2}}, & u_n = \tilde{u}_n/\sqrt{n}.
\end{cases}$$

Here, we introduced a constant $\beta$ to compensate for growth of the sequence $(u_n)$ and can be arbitrarily chosen to control its range of values. The simulations and reconstruction presented here are obtained with the heuristic choice $\beta \approx \exp(-3 \max x)$, where $\max x$ is the maximum obtained homodyne value.
Likewise, the corresponding definition of $v_m$ best suited for numerics follows. It is defined through a forward or backward recursion [11, 18] depending on the value of $x$. The backward recursion can be used if

$$|x| < \alpha_4 - \frac{1}{2} \frac{1}{\sqrt[4]{|x|}}, \quad (8)$$

with $\alpha_n = \sqrt{n + 1/2}$. Then the recursion can be safely started from $m = 4M$:

$$
\begin{align*}
&v_{4M}(x) = \beta^{-1} e^{-x^2} \kappa_{4M}, \\
&\tilde{v}_{4M}(x) = \sqrt{4M}
\end{align*}
\quad (9)
\begin{align*}
&v_{4M-1}(x) = \beta^{-1} e^{-x^2} \kappa_{4M-1}, \\
&\tilde{v}_{4M-1}(x) = \sqrt{4M-1}v_{4M-1}(x), \\
&v_m(x) = \sqrt{\frac{m}{2x}} v_{m-1}(x), \\
&\tilde{v}_m(x) = \sqrt{\frac{m}{2x}} \tilde{v}_{m-1}(x).
\end{align*}

For values of $x$ outside the region of Eq. (8), one should use the forward recursion

$$
\begin{align*}
&v_0(x) = \beta^{-1}, \\
&\tilde{v}_0(x) = 0, \\
&v_m(x) = \sqrt{\frac{m}{2x}} v_{m-1}(x), \\
&\tilde{v}_m(x) = \sqrt{\frac{m}{2x}} \tilde{v}_{m-1}(x). \quad (10)
\end{align*}
$$

We scaled the starting value by $\beta^{-1} e^{-x^2}$ to compensate for the divergence of the vectors $v_m$ and also canceling the effect of the constant $\beta$, which is needed only to change the range of values assumed by the vectors $v_n$ and $v_m$. Now the pattern functions $f_{n,m}$ can be rewritten as

$$f_{n,m}(x) = \left[ 2x u_n(x) - \tilde{u}_{n+1}(x) \right] v_m(x) - u_n(x) \tilde{v}_{m+1}(x). \quad (11)$$

In Fig. 4 some examples of very large reconstructions of density matrices are presented, which use the above recursions.

A. Calculating the Wigner function

As discussed above, the HCT method does not require the reconstruction of the Wigner function, as it directly reconstructs the density matrix (and the expectation value of arbitrary operators). Still, it may be useful to calculate also the Wigner function [19], defined as

$$W(\alpha, \tilde{\alpha}) = \frac{2}{\pi} \mathrm{Tr} \left[ \rho e^{2(\alpha a^\dagger - \tilde{\alpha} \tilde{a}^\dagger)} e^{i \pi a^\dagger a} \right], \quad (12)$$

which can be explicitly written as

$$W(\alpha, \tilde{\alpha}) = \frac{2}{\pi} \Re \sum_{d=0}^{+\infty} e^{i d \phi} \arg \sum_{n=0}^{+\infty} \Lambda_{n,d}(2|\alpha|^2) \rho_{n,n+d}, \quad (13)$$

where $\rho_{n,m} = \langle n|\rho|m \rangle$, and

$$\Lambda_{n,d}(x) = (-1)^n (2 - \delta_{d0}) x^{d/2} \sqrt{\frac{n!}{(n+d)!}} L_n^d(x). \quad (14)$$

Here, $L_n^d$ is defined as $L_n^d(x) = e^{-x/2} L_n^d(x)$, where $L_n^d(x)$ are the (generalized) Laguerre polynomials.

The formula in Eq. (14) does not perform well for lower precision data types as there are several quantities that become very large quite fast. Although one can mitigate this by, for instance, computing first the logarithm of the factorials, this may still be not enough for highly excited states that occupy a large part of the phase space. This problem can be solved with appropriately designed recursion relations that we present in Appendix B. The two methods presented there are inequivalent, but have roughly the same efficiency.

For the numerical implementation we write the Wigner function in polar coordinates $W(r, \vartheta)$, with $\alpha = r e^{i \vartheta}$:

$$W(r, \vartheta) = \Re \sum_{d=0}^{M-1} \frac{e^{i d \vartheta}}{1 + \delta_{d0}} \sum_{n=0}^{M-d-1} \lambda_{n,d}(4r^2) \tilde{\rho}_{n,d}, \quad (15)$$

where $\tilde{\rho}_{n,d} = (-1)^n \rho_{n,n+d}$ and $\lambda_{n,d}$ can be defined recursively leveraging the definition of the Laguerre polynomials, as shown in Appendix B. It is convenient to rewrite the density matrix by diagonals, reorganizing its elements for efficient memory access.
Figure 5: Wigner functions of the simulations presented in Fig. 4. Top: $|\psi\rangle \propto |600\rangle + |700\rangle$; bottom: $|\psi'\rangle \propto |\alpha\rangle + |\alpha\rangle$, with $\alpha = 13$. To calculate these, the optimized formula in Eq. (15) was used. A window has been applied to the images for a better contrast.

III. SOFTWARE PACKAGE

The algorithms presented in this paper have been implemented in a Julia [20–22] package and a C++ package. The repositories are hosted on GitLab under the HomodyneCT group [23]. The C++ package documentation can be found at [24].

The Julia package ecosystem for imaging and quantum tomography under the GitLab group HomodyneCT consists of two packages: MartaCT [25] and HomodyneImaging [26]. The first one implements the tools needed for image analysis and the traditional reconstruction algorithms with FBP methods (this package is open-source and distributed under the MIT license); the second one implements all the quantum algorithms described in this paper (this package is currently closed source, so if you need access to it, please contact the authors).

The main implementation resides in the Julia package HomodyneImaging, along with the companion package MartaCT. The basic interface is defined in MartaCT, while HomodyneImaging extends it, leveraging the multiple dispatch system of the language.

In order to give a sense to the reader of how MartaCT can be used, we provide an example to reconstruct a symmetric cat state $|\psi\rangle \propto |\alpha\rangle + |\alpha\rangle$, with $\alpha = 3$. The example Listing 1 uses the Julia package QuantumOptics [27] to construct the quantum state. The resulting reconstruction is depicted in Fig. 6.

Let us dive in a bit in the source code of the example. First of all, in Julia one needs to import the necessary modules providing the functionality we are going to use: this is done with the using keyword, followed by a comma-separated list of module names. We import several modules: HomodyneImaging for the HCT algo-

Listing 1: Code example for Fig. 6.

```julia
using HomodyneImaging, QuantumOptics, IntervalSets, Plots
using MartaCT.Simulations
M = 64  # density matrix dimension
bs = FockBasis(M-1)
a = 3
norm = sqrt(2*(1+exp(-2*abs2(a))))
psi = (coherentstate(bs, a) + coherentstate(bs, -a)) / norm
rho = tensor(psi, psi')
xs = linspace(-5..5, 600)
W = wigner(rho, xs, xs, FFTWigner())
fs = linspace(ORI(0..2pi), 300)
marg = radon(W, xs, fs, RadonSquare())
sim = SinogramHomodyneSimulation(
    nsamples=1000, nblks=10, ndim=M)
rhosim, rhodiag, yerrors = simulate(sim, marg, xs, fs)
Wsim = wigner(rhosim, xs, xs, FFTWigner())
dmp = bar(0:M-1, rhodiag;
    c=:red,
    yerrors,
    leg=:none,
    lw=0,
    lc=:red,
    xguide=“\$n\$",
    guidefontsize=14,
)
wigp = heatmap(xs, xs, Wsim;
    xguide=“\$x\$",
    yguide=“\$y\$",
    guidefontsize=14,
)
plot(
    dmp,
   wigp;
    dpi=300
)
```

Figure 6: Example image obtained with the HomodyneImaging package representing a cat state $|\psi\rangle \propto |\alpha\rangle + |\alpha\rangle$, with $\alpha = 3$. 
IV. CONCLUSIONS

In conclusion, we have reviewed the homodyne tomography technique, showing how to adapt it to high-dimensional state reconstructions. We have described how this method can be implemented in practice. We gave some illustrative examples in the form of Monte Carlo simulations of the reconstruction experiments. They demonstrate the robustness of the method.

Furthermore, we present the software packages implementing the reconstruction algorithms. The main development has been focused on the Julia packages providing a new promising ecosystem for both medical imaging and quantum tomography applications.

V. ACKNOWLEDGMENTS

This work has been possible thanks to the support of the ATTRACT project “Quantum Imaging for Tomography” (QuIT) https://attract-eu.com/showroom/project/quantum-imaging-for-tomography-quit and also the Unitary Fund project https://unitary.fund/grants.html. LM acknowledges support from the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Superconducting Quantum Materials and Systems Center (SQMS) under contract number DE-AC02-07CH11359.

Appendix A: Review of the derivation of the tomographic formulae

The tomographic reconstruction for the homodyne detector relies on the fact that the displacement operators $D(\alpha) = e^{\alpha a^\dagger - \alpha a}$ are a complete orthonormal basis for the space of operators with respect to the Hilbert–Schmidt scalar product: $\langle A|B \rangle = \text{Tr}[A^\dagger B]$. More precisely, for any linear operator $A$ one can write

$$A = \frac{1}{\pi} \int_{\mathbb{C}} \text{Tr}[AD^\dagger(\alpha)]D(\alpha) d^2\alpha = \frac{1}{4\pi} \int_0^\pi \int_{\mathbb{R}} \text{Tr}[Ae^{ikX_\phi}]e^{-ikX_\phi}|k\rangle dk d\phi,$$

(A1)

where $\alpha$ is written in polar coordinates in the last equation: $\alpha = ike^{i\phi}/2$, and the first integral can be taken only on the interval $[0, \pi]$ because of the symmetry $X_{\phi+\pi} = -X_\phi$.

The tomographic formula is then obtained by introducing the probability $p_\phi(x) = \langle x|\rho|x \rangle_\phi$ of getting $x$ when measuring $X_\phi$:

$$\langle A \rangle = \text{Tr}[A\rho] = \frac{1}{\pi} \int_0^\pi \int_{\mathbb{R}} p_\phi(x) K_\phi[A](x) dx d\phi. \quad (A2)$$

The operator $K_\phi[A]$ is the so-called kernel of homodyne tomography, defined by

$$K_\phi[A](x) = \frac{1}{4} \int_{\mathbb{R}} \text{Tr}[Ae^{ik(X_\phi-x)}]|k\rangle dk. \quad (A3)$$

Now, thanks to the trace in the integral, the kernel does not necessarily diverge. It is possible to classify the set operators $A$ which produce a bounded kernel [28] in Eq. (A2). In the simplest case considered here, the operator $A$ should be at least Hilbert–Schmidt. In particular this is the case of $|n\rangle\langle m|$ which provides the matrix elements $\rho_{nm}$.

Since the first appearance [8] of this method there have been several attempts to simplify the expression of the tomographic formula for $\rho_{n,m}$. Let us rewrite Eq. (A2) for $\rho_{n,m}$ with explicit phase dependence:

$$\rho_{n,n+d} = \frac{1}{\pi} \int_0^\pi d\phi e^{-id\phi} \int_{\mathbb{R}} dx p_\phi(x) f_{n,n+d}(x), \quad (A4)$$

where $f_{n,m}(x)$ are the so-called pattern functions, which are real and satisfy the symmetries:

$$f_{n,m}(x) = f_{m,n}(x), \quad (A5)$$

$$f_{n,m}(-x) = (-1)^{n-m} f_{n,m}(x). \quad (A6)$$

The analytical expression for the pattern functions $f_{n,m}(x)$ has been obtained by D’Ariano, Leonhardt and Paul [9–11] symplifying previous derivations as follows (notation
and conventions from [19]):

\[ f_{n,m}(x) = 2 \sqrt{\frac{n!}{(n+d)!}} e^{-x^2}. \]

\[ \sum_{\nu=0}^{n} \frac{(-1)^\nu}{\nu!} \frac{(n+d)!}{(n-\nu)!(n-\nu+1)!} \cdot \left( -i \right)^d D_{-(2\nu+d+2)}(-2ix) \],  

(A7)

where \( D_\lambda(z) \) denotes the parabolic cylinder function. Such expression is very delicate to be used in numerical implementations for large quantum numbers.

Further simplifications were possible thanks to an insight of Richter [29] who has been able to link the tomographic formula to the regular and irregular solutions of the Schrödinger equation of the harmonic oscillator. The pattern functions are related to the solutions of the Schrödinger equations employing an Hilbert transformation. The pattern functions \( f_{n,m} \) can be written as a derivative of some functions \( g_{n,m} \) [11, 18]:

\[ f_{n,m}(x) = \frac{\partial g_{n,m}}{\partial x}(x), \]  

(A8)

where the functions \( g_{n,m} \) are obtained through a Hilbert transformation:

\[ g_{n,m}(x) = \frac{\mathcal{P}}{\pi} \int_{\mathbb{R}} \frac{u_n(\xi) u_m(\xi)}{x - \xi} \, d\xi, \]  

(A9)

where \( \mathcal{P} \) denotes the Cauchy principal value and \( u_n \) are the normalizable solutions of the Schrödinger equation

\[ \left(-\frac{1}{2} \frac{\partial^2}{\partial x^2} + \frac{x^2}{2}\right) u_n(x) = \left(n + \frac{1}{2}\right) u_n(x). \]  

(A10)

In the same ref. [18], the authors were able to obtain an explicit expression for the functions \( g_{n,m} \):

\[ g_{n,m}(x) = u_n(x) v_m(x), \quad \text{for } m \geq n, \]  

(A11)

while for \( m < n \) one can use \( g_{n,m} = g_{m,n} \). Here, the functions \( v_m \) are the irregular solutions of the Schrödinger equation, namely unnormalizable solutions continued to the complex plane:

\[ \left(-\frac{1}{2} \frac{\partial^2}{\partial z^2} + \frac{z^2}{2}\right) v_m(z) = \left(m + \frac{1}{2}\right) v_m(z), \]  

(A12)

with \( v_m(z) \) real on the real axis \( z = x \).

The evaluation of the pattern functions \( f_{n,m} \) can be implemented efficiently on a computer by employing the factorized form

\[ f_{n,m}(x) = 2x u_n(x) v_m(x) - \frac{\sqrt{n+1} u_{n+1}(x) v_m(x) - \sqrt{m+1} u_n(x) v_{m+1}(x)}{\sqrt{m+1} u_n(x)}(A13) \]

for \( m \geq n \), while for \( m < n \) we use the symmetry \( f_{n,m}(x) = f_{m,n}(x) \). On one side, for the “regular” solutions \( u_n(x) \) one can use the following recurrence relation:

\[ \begin{cases} 
  u_0(x) = e^{-x^2} \\
  u_1(x) = 2xe^{-x^2} \\
  u_n(x) = \frac{1}{\sqrt{n}} (2x u_{n-1}(x) - \sqrt{n-1} u_{n-2}(x))
\end{cases} \]  

(A14)

On the other side, for the irregular wavefunctions \( \varphi_m \), we have to setup a more careful construction, as explained in [18]. In the region given by the Bohr–Sommerfeld radius, the classically allowed region, we should employ a backward recursion instead. A safe choice for this is to consider the region

\[ |x| < \alpha_{4k} - \frac{1}{2} \frac{1}{\sqrt{\alpha_{4k}}}, \]  

(A15)

with \( \alpha_m = \sqrt{n+1/2} \). In this case, it is recommended to use the backward recursion as follows:

\[ v_m(x) = \frac{1}{\sqrt{m+1}} (2x v_{m+1}(x) - \sqrt{m+2} v_{m+2}(x)), \]  

(A16)

with initial values obtained from a semiclassical approximation [18] valid for large \( m \)

\[ v_m(x) = \kappa_m := \frac{(8\pi)^{1/4}}{\sqrt{\alpha_m} \sin \tau_m} \sin \left( \frac{1}{2} \alpha_m \chi_m + \frac{\pi}{4} \right), \]  

(A17)

where

\[ \tau_m = \arccos(x/\alpha_m), \quad \chi_m = \sin(2\tau_m) - 2\tau_m. \]

When \( x \) is outside of the region Eq. (A15), one can employ the asymptotic form for \( v_m \), which is implemented by a forward recursion [18]:

\[ \begin{cases} 
  v_0(x) = \frac{1}{2} e^{x^2} \\
  v_m(x) = \frac{\sqrt{m}}{2\pi} v_{m-1}(x)
\end{cases} \]  

(A18)

**Appendix B: Recurrences for the Wigner function**

We have seen an optimized form of the reconstruction method for the density matrix, but it may still be interesting to recover also the Wigner function. We present here two recursive methods to compute the functions \( \lambda_{n,d} \), leveraging the recurrence relations of the Laguerre polynomials \( L_n^d(x) \).

**1. Method 1**

Let us first recall the recursive definition of the (generalized) Laguerre polynomials [30]:

\[ L_0^d(x) = 1, \]
\[ L_1^d(x) = 1 + d - x, \]
\[ L_n^d(x) = a_{n,d} L_{n-1}^d(x) - b_{n,d} L_{n-2}^d(x), \]  

(B1)

where

\[ a_{n,d} = \frac{n + d}{n + 1} \quad \text{and} \quad b_{n,d} = \frac{n - d}{n + 1}. \]
with
\[ a_{n,d} = \frac{2n + d - x - 1}{n}, \quad b_{n,d} = \frac{n + d - 1}{n}. \]

We introduce the function
\[ G_n^d(x) = \sqrt{\frac{n!}{(n + d)!}} L_n^d(x), \quad (B2) \]

where \( L_n^d(x) = z(x)L_n^d(x) \) and \( z(x) = \Lambda_0 e^{-x^2} \), with \( \Lambda_0 = 4/\pi \). The function \( \lambda_{n,d}(x) \) in Eq. (15), thus, can be computed as
\[ \lambda_{n,d}(x) = x^{d/2} G_n^d(x). \quad (B3) \]

In the following we will drop the dependence on the variable \( x \). \( \lambda_{n,d} \) can be viewed as a matrix of which only the elements \( n \leq M - d - 1 \) have to be computed:
\[
\begin{array}{ccccccc}
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\end{array}
\]

The starting point is \( \lambda_{0,0} = z \), and the first 2 rows can be obtained by
\[ \lambda_{0,d} = \sqrt{\frac{z}{d}} \lambda_{0,d-1}, \quad \lambda_{1,d} = \frac{1 + d - x}{\sqrt{d+1}} \lambda_{0,d}, \quad (B4) \]
while the first column \( \lambda_{n,0} = L_n^0 \) is just given by the (scaled) Laguerre polynomials.

At this point we can provide the general recursive formula, valid for \( n \geq 2 \) and \( d \geq 0 \) (for \( d = 0 \) the following relation just simplifies to that of the Laguerre polynomials, so that the first column can be computed with specialized formulae):
\[ \lambda_{n,d} = a'_{n,d} \lambda_{n-1,d} - b'_{n,d} \lambda_{n-2,d}, \quad (B5) \]

where
\[ a'_{n,d} = \frac{2n + d - x - 1}{\sqrt{n(n + d)}}, \]
\[ b'_{n,d} = \frac{(n - 1)(n + d - 1)}{n(n + d)}. \]

2. Method 2

We can provide another implementation of the Eq. (15) computing the function \( \lambda_{n,d} \) with different recurrence relations which rely on the summation property of the Laguerre polynomials \( L_n^d \) which extends to the functions \( G_n^d(x) \):
\[ L_n^d = L_{n-1}^d + L_{n-1}^{d-1}. \quad (B6) \]

Again the starting point of the recursion is \( \lambda_{0,0} = z \) and the first row is given by \( \lambda_{0,d} = \sqrt{\frac{z}{d}} \lambda_{0,d-1} \), while the first column is obtained as before from \( L_n^0 \):

\[
\begin{array}{ccccccc}
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\
\end{array}
\]

The general recursion formula, valid for \( n \geq 1 \) and \( d \geq 1 \), is now given by:
\[ \lambda_{n,d} = \sqrt{n-1,d} - \sqrt{n,d} \lambda_{n,d-1}, \quad (B7) \]

With respect to the first method, in this case one needs an additional vector to store the values \( \lambda_{n,d-1} \).

[1] G. M. D’Ariano and H. P. Yuen, Phys. Rev. Lett. 76, 2832 (1996).
[2] G. M. D’Ariano, L. Maccone, and M. G. A. Paris, Journal of Physics A: Mathematical and General 34, 93 (2000).
[3] G. M. D’Ariano, M. G. A. Paris, and M. F. Sacchi, Advances in Imaging and Electron Physics 128, 206 (2003).
[4] G. M. D’Ariano, M. G. Paris, and M. F. Sacchi, 2 quantum tomographic methods, in Quantum State Estimation, edited by M. Paris and J. Řeháček (Springer Berlin Heidelberg, Berlin, Heidelberg, 2004) pp. 7–58.
[5] G. M. D’Ariano, L. Maccone, and M. F. Sacchi, in Quantum Information With Continuous Variables of Atoms.
and Light (World Scientific, 2007) pp. 141–158.
[6] L. G. Lutterbach and L. Davidovich, Phys. Rev. Lett. 78, 2547 (1997).
[7] K. Vogel and H. Risken, Phys. Rev. A 40, 2847 (1989).
[8] G. M. D’Ariano, C. Macchiavello, and M. G. A. Paris, Phys. Rev. A 50, 4298 (1994).
[9] H. Paul, U. Leonhardt, and G. M. D’Ariano, Acta Physica Slovaca 45, 261 (1995).
[10] G. M. D’Ariano, U. Leonhardt, and H. Paul, Phys. Rev. A 52, R1801 (1995).
[11] U. Leonhardt, H. Paul, and G. M. D’Ariano, Phys. Rev. A 52, 4899 (1995).
[12] R. J. Rossi, Mathematical statistics: an introduction to likelihood based inference (John Wiley & Sons, 2018).
[13] B. Vlastakis, G. Kirchmair, Z. Leghtas, S. E. Nigg, L. Frunzio, S. M. Girvin, M. Mirrahimi, M. H. Devoret, and R. J. Schoelkopf, Science 342, 607 (2013).
[14] C. Eichler, D. Bozyigit, C. Lang, L. Steffen, J. Fink, and A. Wallraff, Phys. Rev. Lett. 106, 220503 (2011).
[15] C. Eichler, D. Bozyigit, and A. Wallraff, Phys. Rev. A 86, 032106 (2012).
[16] F. Mallet, M. A. Castellanos-Beltran, H. S. Ku, S. Glancy, E. Knill, K. D. Irwin, G. C. Hilton, L. R. Vale, and K. W. Lehnert, Phys. Rev. Lett. 106, 220502 (2011).
[17] U. Leonhardt and M. Munroe, Phys. Rev. A 54, 3682 (1996).
[18] U. Leonhardt, M. Munroe, T. Kiss, T. Richter, and M. Raymer, Optics Communications 127, 144 (1996).
[19] G. M. D’Ariano, Measuring quantum states, in Quantum Optics and the Spectroscopy of Solids: Concepts and Advances, edited by T. Hakoğlu and A. S. Shumovsky (Springer Netherlands, Dordrecht, 1997) pp. 175–202.
[20] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, arXiv preprint arXiv:1209.5145 (2012).
[21] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, SIAM Review 59, 65 (2017).
[22] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, Julia language (2012).
[23] N. Mosco, Homodyne computed tomography (2019).
[24] N. Mosco, Hct tools (2019).
[25] N. Mosco, Marta ct (2019).
[26] N. Mosco, Homodyne imaging (2019).
[27] S. Krämer, D. Plankensteiner, L. Ostermann, and H. Ritsch, Computer Physics Communications 227, 109 (2018).
[28] G. M. D’Ariano, in In International School of Physics Enrico Fermi (IOS Press, 2002).
[29] T. Richter, Physics Letters A 211, 327 (1996).
[30] M. Abramowitz, I. A. Stegun, and R. H. Romer, Handbook of mathematical functions with formulas, graphs, and mathematical tables (American Association of Physics Teachers, 1988).