Scalable quantum detector tomography by high-performance computing

Timon Schapeler[1,2*†], Robert Schade[3†], Michael Lass[3,4], Christian Plessl[3,4], Tim J. Bartley[1,2]

[1]Department of Physics, Paderborn University, Warburger Str. 100, 33098 Paderborn, Germany.
[2]Institute for Photonic Quantum Systems (PhoQS), Paderborn University, Warburger Str. 100, 33098 Paderborn, Germany.
[3]Paderborn Center for Parallel Computing, Paderborn University, Warburger Str. 100, 33098 Paderborn, Germany.
[4]Department for Computer Science, Paderborn University, Warburger Str. 100, 33098 Paderborn, Germany.

*Corresponding author(s). E-mail(s): timon.schapeler@uni-paderborn.de;
Contributing authors: robert.schade@uni-paderborn.de;
†These authors contributed equally to this work.

Abstract

At large scales, quantum systems may become advantageous over their classical counterparts at performing certain tasks. Developing tools to analyse these systems at the relevant scales, in a manner consistent with quantum mechanics, is therefore critical to benchmarking performance and characterising their operation. While classical computational approaches cannot perform like-for-like computations of quantum systems beyond a certain scale, classical high-performance computing (HPC) may nevertheless be useful for precisely these characterisation and certification tasks. By developing open-source customised algorithms using high-performance computing, we perform quantum tomography on a megascale quantum photonic detector covering a Hilbert space of $\mathbf{10^6}$. This requires finding $\mathbf{10^8}$ elements of the matrix corresponding to the positive operator valued measure (POVM), the quantum description of the detector, and is achieved in minutes of computation time. Moreover, by exploiting the structure of the problem, we achieve highly efficient parallel scaling, paving the way for quantum objects up to a system size of $\mathbf{10^{12}}$ elements to be reconstructed using this method. In general, this shows that a consistent quantum mechanical

description of quantum phenomena is applicable at everyday scales. More concretely, this enables the reconstruction of large-scale quantum sources, processes and detectors used in computation and sampling tasks, which may be necessary to prove their nonclassical character or quantum computational advantage.

Keywords: scaling, reconstruction, quantum detector tomography, single-photon detector, high-performance computing, Wigner functions, parallelisation, quantum photonics

Photonic quantum computing paradigms are built around large scale generation, manipulation and measurement of quantum light. At sufficient scale, the computations these devices perform cannot be verified by conventional computing. Pertinent examples are quantum simulators [1] and Boson sampling [2], where in the latter beyond a certain system size, the process of sampling from the output distribution of a nonclassical input state in a photonic circuit implementing a random unitary matrix is a task which is computationally easier for a photonic quantum processor. Nevertheless, the same photonic processor, under illumination from "classical" light, is computationally "easy" to compute using classical approaches. Since this classical light is tomographically complete, one can use techniques such as quantum tomography to characterize the device and verify its underlying quantum mechanical structure, without performing the full Boson sampling task [3]. However, just because the computational complexity class suggests that the problem is "easy" for a classical computer, still the question arises what the practical limits of this approach are. To that end, high-performance computing (HPC) is a very well established field which has great potential to assist in these quantum tomography tasks, provided the benefits of parallelisation can be reconciled with the constraints imposed by the quantum mechanical objects to be reconstructed.

Quantum detector tomography [4–9] is a well-established technique for providing a consistent quantum mechanical characterisation of the measurement process. This approach to characterising experiments is particularly attractive, since it provides a model-free method to connect the underlying quantum mechanics of systems to the measurement results we observe. The aim of a tomography experiment is to reconstruct the set of Positive Operator Valued Measures (POVMs) $\{\boldsymbol{\pi}_n\}$ by mapping the detector response to a tomographically complete set of input states, i.e. the set of input states that span the full outcome space of the detector. The size of the problem is governed by the dimensionality of the Hilbert space M occupied by the set of input states and the number of outcomes N. In order to be tomographically complete, the Hilbert space spanned by the input states is necessarily at least as large as the outcome space, i.e. $M \gtrsim N$. In general, the size of the set of POVMs $\{\boldsymbol{\pi}_n\}$ is then $M^2 \cdot N$; for non-phase-sensitive detectors this reduces to $M \cdot N$. Nevertheless, the challenge is thus to devise techniques to reconstruct POVMs covering ever larger system sizes, to enable state of the art quantum optics experiments [3, 10].

Up to now, almost all detector tomography experiments have described detectors with few outcomes ($N \lesssim 10$) covering a relatively small Hilbert space ($M \lesssim$

100) [7, 8, 11–17], where approaches such as semi-definite programming and maximum likelihood estimation could be readily applied with standard computing hardware. Alternatively, data pattern tomography can be used to characterise a relevant subset of the detector's outcome space [18]. More recently, numerical approaches using convex optimisation solvers have been pursued, to interrogate larger system sizes [19]. This approach has been applied to high-performance computing hardware, as investigated by Liu et al., [20] using simulated data. Their results suggested an upper limit of system size $M \cdot N$ of the order 10^5, based on available computational resources. In the case of phase sensitive detectors, the size of the matrix required to map the Hilbert space dimension M becomes M^2, significantly increasing the computational resource requirements. In this context, the largest tomographic reconstruction to date has been performed on a phase-sensitive photon counter, requiring the reconstruction of $1.8 \cdot 10^6$ elements [21].

In this work, we perform experimental detector tomography of a high dynamic range detector up to $N = 50$ occupied outcomes covering a Hilbert space of $M = 1.2 \cdot 10^6$ photons, surpassing the limit suggested by Liu et al. [20] as well as the size of the experimental reconstruction by Zhang et al. [21], by two orders of magnitude. To do so, we developed a convex minimisation solver optimised for operation on high-performance computing hardware. We apply this solver to the simulated data from Liu et al. (340 outcomes, Hilbert space up to $3 \cdot 10^4$ photons), and show two orders of magnitude saving in runtime, and four orders of magnitude saving in memory usage. Extending the simulated data further, we demonstrate reconstructions of system sizes $> 10^{12}$ (e.g. 10^6 outcomes, 10^6 photons). Among many other applications, this allows for a quantum characterisation of state-of-the-art single-photon sensitive detector arrays [10, 22] and Boson sampling machines [3], in minutes of computation time.

1 Results

1.1 Tomographic approach

The principle of operation of detector tomography is depicted in Fig. 1. Given a tomographically complete set of input states (beige), many measurements are performed, resulting in a set of outcome probabilities (green) related to the outcomes via the set of POVMs (red). Quantum mechanically, measurement outcomes $p_{\boldsymbol{\rho},n}$ on each state $\boldsymbol{\rho}$ via the POVM $\boldsymbol{\pi}_n$ are governed by the Born rule $p_{\boldsymbol{\rho},n} = \mathrm{Tr}\left[\boldsymbol{\rho}\boldsymbol{\pi}_n\right]$. Taking all measurements together, one can recast the Born rule as the matrix equation

$$\boldsymbol{P} = \boldsymbol{F}\boldsymbol{\Pi} \tag{1}$$

where the matrix of outcome probabilities $\boldsymbol{P}_{D\times N}$ (green) are related to the set of input states $\boldsymbol{F}_{D\times M}$ (beige) via the set of POVMs $\boldsymbol{\Pi}_{M\times N}$ (red). The experimental setup used to generate and measure the set of input states is shown in Fig. 1, with experimental details given in the Methods section.

In principle, one can invert Eq. (1) to recover the unknown matrix of POVMs $\boldsymbol{\Pi}$. In practice, however, the direct inversion of Eq. (1) often yields nonphysical results,

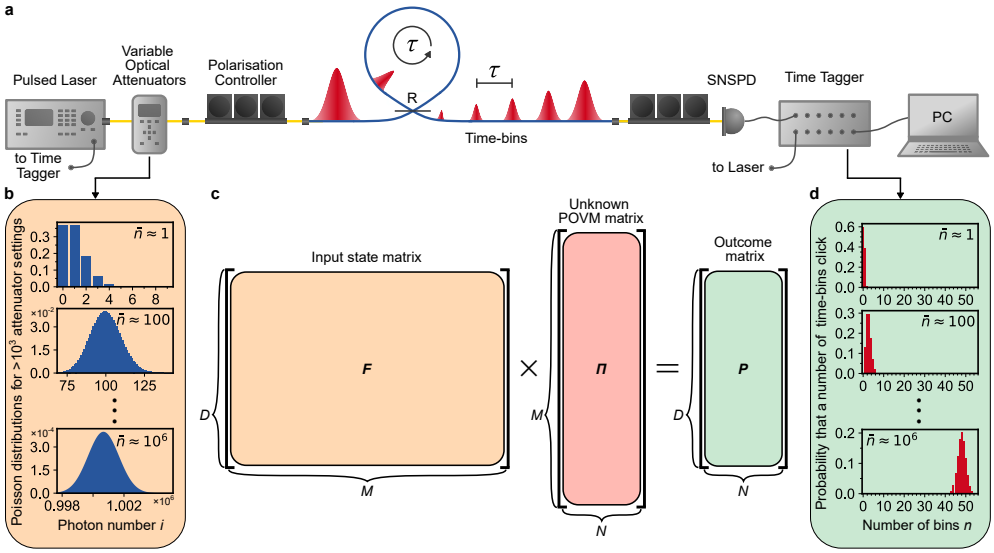

Fig. 1 (a) Experimental setup to perform high dynamic range quantum detector tomography. The coherent states from a picosecond pulsed laser can be attenuated with variable optical attenuators, to control their mean photon numbers. Polarisation controllers before the fiber beam splitter loop (blue line) and the SNSPD are used to optimize the performance of the devices. The beam splitter loop creates sub-pulses with a temporal separation of $\tau = 156$ ns and has an adaptable out-coupling R and loop-efficiency η_{loop}. A time tagger records raw time tags of the electrical output signal of the SNSPD. (b) Poisson distributions for mean photon numbers of $\bar{n} = 1$, $\bar{n} = 100$ and $\bar{n} = 10^6$, which are set by the variable optical attenuator. (c) Schematic matrix representation of the matrices $\boldsymbol{F}_{D\times M}$ containing the coherent input states, $\boldsymbol{\Pi}_{M\times N}$ containing the (unknown) POVMs of the detector and $\boldsymbol{P}_{D\times N}$ containing the measured outcomes of the detector. (d) Probability distributions that a certain number of time-bins of the detector click, i.e., the outcomes of the detector for mean photon numbers of $\bar{n} = 1$, $\bar{n} = 100$ and $\bar{n} = 10^6$.

i.e. POVMs which are not positive semi-definite and/or unit trace. This is due to the presence of unavoidable noise in the experimental outcomes. To overcome this, one can recast Eq. (1) into the constrained minimisation problem

$$\min ||\boldsymbol{P} - \boldsymbol{F}\boldsymbol{\Pi}||_2 \,, \tag{2a}$$

$$\text{subject to } \boldsymbol{\pi}_n \succeq 0 \,, \quad \sum_{n=0}^{N-1} \boldsymbol{\pi}_n = \boldsymbol{I} \,, \tag{2b}$$

which searches for the set of parameters in the POVM matrix $\boldsymbol{\Pi}$ which, when multiplied by the set of input states $\boldsymbol{F}$, most closely match the measured data $\boldsymbol{P}$. The two-norm, $||\boldsymbol{A}||_2 = \sqrt{\sum_{ij} A_{i,j}^2}$, is used and $\boldsymbol{A} \succeq 0$ denotes the positive semi-definiteness constraint of the matrix $\boldsymbol{A}$.

The solution of the minimisation problem Eq. (2a) with the constraints Eq. (2b) for large POVM matrices is a numerically challenging convex optimization problem. Furthermore, the structure of the problem and the constraints render parallelisation

nontrivial, since an efficient communication scheme between nodes is required. The solution's computational effort and memory usage mainly depend on the number of free variables, i.e. the size of the matrix of POVMs $\mathbf{\Pi}$. The construction of $\mathbf{\Pi}$ from $\boldsymbol{\pi}_n$, and therefore the matrix dimensions, depend on whether or not the detector is sensitive to coherence present in the input states. In what follows, we consider phase insensitive detectors, the POVMs $\boldsymbol{\pi}_n$ of which can be expressed as diagonal matrices, with elements $(\pi_n)_{k,l}$ nonzero for $k = l$. For each $\boldsymbol{\pi}_n$, these nonzero elements comprise each row of the matrix of POVMs $\mathbf{\Pi}$, i.e. $\Pi_{i,n} = (\pi_n)_{k=i,l=i}$. In this case, the size of the matrix and the number of entries in the matrix $\mathbf{\Pi}$ is thus $M \cdot N$, where $M - 1$ denotes the maximal photon number in the Hilbert space with dimension M, and N is the number of detector outcomes. Furthermore, in this case the constraints on $\boldsymbol{\pi}_n$ are such that the elements of $\mathbf{\Pi}$ are non-negative. The constrained minimisation problem in the conventional form is thus

$$\min_{\mathbf{\Pi} \in \mathbb{R}^{M \times N}} \quad ||\boldsymbol{P} - \boldsymbol{F}\mathbf{\Pi}||_2^2 \tag{3a}$$

$$\Pi_{i,n} \geq 0 \; \forall \; i \in \{0, ..., M-1\}, \quad n \in \{0, ..., N-1\} \tag{3b}$$

$$\sum_{n=0}^{N-1} \Pi_{i,n} = 1 \quad \forall \; i \in \{0, ..., M-1\} \tag{3c}$$

for a given $\boldsymbol{P} \in \mathbb{R}^{D \times N}$ and given $\boldsymbol{F} \in \mathbb{R}^{D \times M}$. D denotes the number of probe states. The memory needed for storing $\mathbf{\Pi}$, $\boldsymbol{F}$ and $\boldsymbol{P}$ is roughly

$$\text{mem}_{\text{storage}} = 8 \text{ byte} \cdot (MN + DM + DN). \tag{4}$$

The recent work by Liu et al. [20] has shown the practical solution to this problem for the order of 10^5 free variables using the general minimisation solver MOSEK [23] via CVXPY [24, 25]. The detector setup in their work requires M and D for a given number of outcomes N to be approximately

$$M_{\text{Liu}} \approx 6.6 \cdot N^{1.06} \tag{5}$$

$$D_{\text{Liu}} \approx 4.0 \cdot N^{1.14}. \tag{6}$$

Thus, the memory usage in the situation of Liu at al. for storing $\mathbf{\Pi}$, $\boldsymbol{F}$ and $\boldsymbol{P}$ is

$$\text{mem}_{\text{storage}} \lesssim 2.6 \cdot 10^{-7} \cdot N^{2.2} \text{ GiB}. \tag{7}$$

Using our high-performance computing hardware [26], we investigate the practical main memory usage (see Appendix A for hardware and memory usage benchmark) of the solution approach of Liu et al. [20] for the standard detector tomography (SDT) problem and the simplified modified detector tomography (MDT) problem. Fitting the measured memory usage to the model $\text{mem} = aN^b$ results in

$$\text{mem}_{\text{Liu,SDT}} \approx 1.8 \cdot 10^{-5} \cdot N^{2.95} \text{ GiB} \tag{8}$$

$$\text{mem}_{\text{Liu,MDT}} \approx 2.0 \cdot 10^{-5} \cdot N^{2.93} \text{ GiB} \; . \tag{9}$$

This differs slightly from the scaling approximation given in Ref. [20]; on the one hand, we track the memory usage of the solver continually, and on the other, we use a wider range of total outcomes $N = 11, 21, ..., 201$.

Notwithstanding these slight differences, the prefactors in the memory usage scaling of the solver of Liu et al., Eq. (8)-(9), and the memory needed for the storage of matrices, Eq. (7), are different by two orders of magnitude and the solver exhibits a scaling in N that is significantly higher than for storage. This strongly suggests that a suitable numerical solver with a much lower memory footprint can be constructed to solve this problem more efficiently and scalable.

The goal of our approach is, on the one hand, a memory usage that is only a few times larger than the memory required for storing $\mathbf{\Pi}$, $\boldsymbol{F}$ and $\boldsymbol{P}$ and, on the other hand, the possibility not only to use multiple CPU-cores of a single compute node, but to scale to very large problems by using the distributed memory of many compute nodes. For this purpose, we propose a two-stage variant of the two-metric projected Newton approach [27–29] for the problem in Eq. (3). The algorithm is described in detail in Appendix B.

1.2 Detector under test

We apply our solver to an experimental tomographic dataset obtained from a high dynamic range optical detector, with sensitivity from the single-photon level up to bright light [30]. The high dynamic range is achieved using a multiplexing scheme in which an incoming optical pulse is split into sub-pulses of exponentially decreasing pulse energies, incident on a superconducting single-photon detector (as shown schematically in Fig. 1(a)). The number of outcomes of the detector is governed by the number of sub-pulses which result in a measurement event. Since this detector has no intrinsic phase sensitivity, the POVMs are fully described by matrices diagonal in the number basis, given by $\boldsymbol{\pi}_n = \sum_{k=0}^{M-1} \theta_k^{(n)} |k\rangle\langle k|$. The aim of the reconstruction is thus to find all $\theta_k^{(n)}$ for the different detector outcomes n, up to a maximum photon number $M-1$, corresponding to a Hilbert space of size M.

To carry out the reconstruction, we need a set of input states which span the Hilbert space to which the detector is sensitive. For this task, it is convenient to use the set of coherent states $|\alpha\rangle = \sum_{i=0}^{\infty} e^{-\frac{|\alpha|^2|}{2}} \frac{\alpha^i}{\sqrt{i!}} |i\rangle$. These states provide a photon number distribution governed by Poissonian statistics defined by their mean photon number $|\alpha|^2$. We require a set of D states of different mean photon number $|\alpha_d|^2$, such that the elements of the input state matrix may be written as

$$F_{d,i} = \langle i|\alpha_d\rangle\langle\alpha_d|i\rangle = \frac{|\alpha_d|^{2i}}{i!} e^{-|\alpha_d|^2} \tag{10}$$

for all photon numbers $i \in [0, M-1]$. We limit ourselves to the diagonal elements of the density matrix $|\alpha\rangle\langle\alpha|$ since our detector is insensitive to phase.

While in principle a single mean photon number is sufficient to cover any Hilbert space, since the coefficients are nonzero for all photon numbers i, in practice one requires sufficient statistics for all photon numbers for a reliable reconstruction. We choose mean photon numbers to scale quadratically, i.e., $|\alpha_d|^2 \approx d^2$.

Previously, we have conducted detector tomography of 11 outcomes of this device up to a Hilbert space of size $M \approx 5 \times 10^3$ [19], limited by standard computational methods using CVXPY [24, 25]. Nevertheless, the logarithmic response of the detector means that the Hilbert space to which it is sensitive goes far beyond this limit. Furthermore, the detector design is such that a relatively simple model of the device can be constructed [30], which allows the POVMs to be derived analytically. This enables us to test the accuracy of the computational reconstruction up to arbitrary size, which is essential to ensure correct reconstruction with our solver.

1.3 POVM reconstruction

We investigate the proposed reconstruction method with the experimental measurement data, i.e., with a Hilbert space cutoff of $M = 1210581$, $N = 151$ outcomes, of which the first $\sim$50 outcomes are tomographically covered by the $D = 1076$ input states. Figure 2 shows three different tomographic reconstructions, each of which take different approaches to regularisation (see Methods, Regularisation and smoothing, for details on the regularisation routine). In Fig. 2(a) no regularisation is used, resulting in somewhat noisy response, particularly at low outcome numbers. In Fig. 2(b), a regularisation factor of $\gamma = 10^{-5}$ is used, limited to smoothing between nearest-neighbour photon numbers, as discussed in Ref. [8]. This successfully smooths the high-frequency noise at low photon numbers, but is insufficient (and indeed increases noise) at high photon numbers.

To mitigate this, we introduce a further step to the regularisation which encourages smoother, more physical results at higher photon numbers (see Methods, Regularisation and smoothing). The result of this is shown in Fig. 2(c).

To evaluate the solver in general, and each approach to regularisation in particular, we compare each reconstruction to the analytic model of the POVMs of the detector, shown in Fig. 2(d). In the inset of Fig. 2(d), we plot the infidelity (1-fidelity)

$$F'_n = 1 - F_n = 1 - \frac{\mathrm{Tr}\left(\left(\sqrt{\boldsymbol{\pi}}\boldsymbol{\pi}_{n,\mathrm{theo}}\sqrt{\boldsymbol{\pi}}\right)^{1/2}\right)^2}{\mathrm{Tr}(\boldsymbol{\pi})\mathrm{Tr}(\boldsymbol{\pi}_{n,\mathrm{theo}})} \tag{11}$$

between the reconstructions and the analytic model. Each case shows excellent agreement between the reconstructed and analytical POVMs, with fidelities exceeding 99% for all occupied outcomes in all three cases. For the extended regularisation, this fidelity increases to an average of 99.69% for all occupied outcomes up to $N = 50$. This demonstrates the unparalleled accuracy of the reconstruction at unprecedented Hilbert space size.

From the POVMs of the detector we can reconstruct the Wigner functions corresponding to different detector outcomes, as shown in Fig. 3. These Wigner functions give additional insight about the detector. Negativity at the origin in Fig. 3(b), shows the non-classical nature of the corresponding operator $\boldsymbol{\pi}_{n=1}$. Our approach (see Methods, Computation of Wigner functions) for calculating the Wigner functions is stable even up to a Hilbert space dimension of $M = 10^6$, as can be seen by the smooth nature

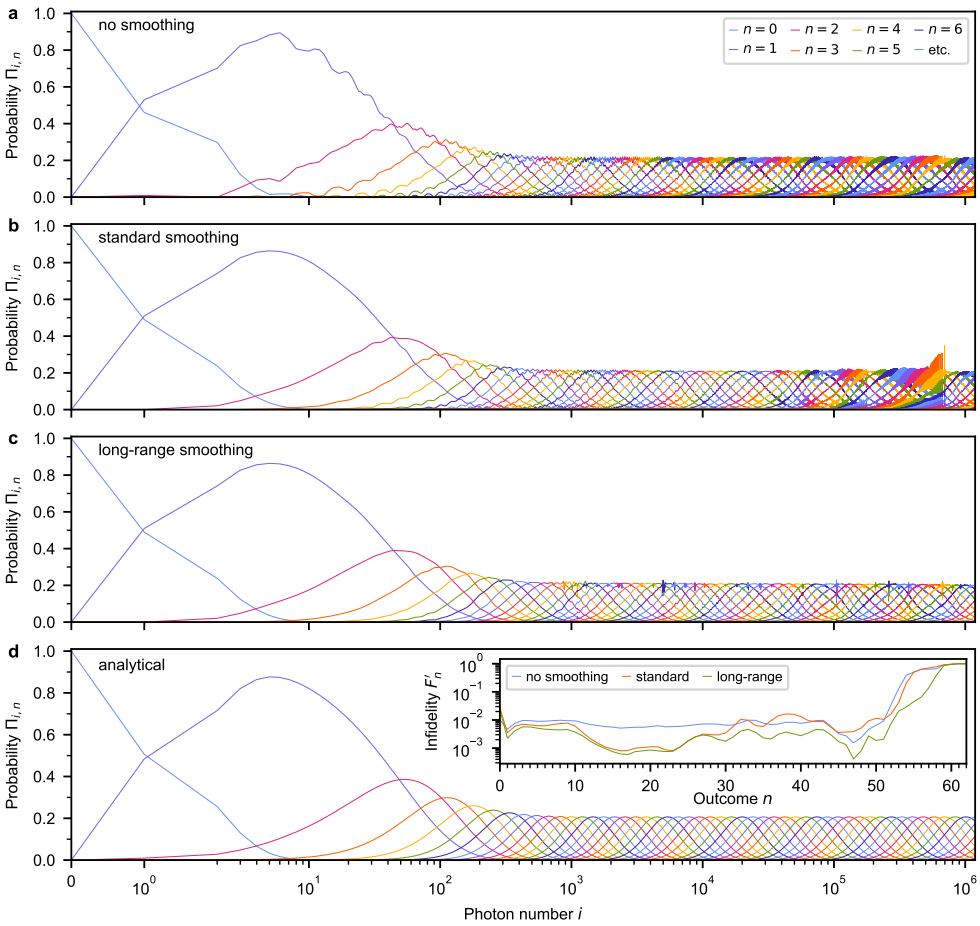

Fig. 2 (a-c) Reconstructed POVMs from the experimentally measured data for $M = 1210581$, $N = 151$ outcomes, and $D = 1076$. Due to the span of the input space, the first ∼50 outcomes are occupied and shown. (a) Does not include regularisation (smoothing) of the POVMs, (b) uses the standard nearest-neighbour smoothing with a regularisation parameter of $\gamma = 10^{-5}$ and (c) in addition to the nearest-neighbour smoothing, utilises a novel long-range approach to the regularisation of the POVMs (see Methods, Regularisation and smoothing, for further detail). (d) Shows the analytical POVMs of the detector. The inset shows the infidelity Eq. (11) between the three regularisation approaches and the analytical model for all occupied outcomes of the detector.

of the Wigner functions of the analytical POVMs in Fig. 3 (black dashed lines). Consequently, the noise in the Wigner function of the experimental POVM $\boldsymbol{\pi}_{n=40}$ can be explained by noise in the reconstructed POVM (which is visible in Fig. 2(c)). However, we see that the general overlap between Wigner functions of experimental and analytical POVMs is high, especially for small outcomes.

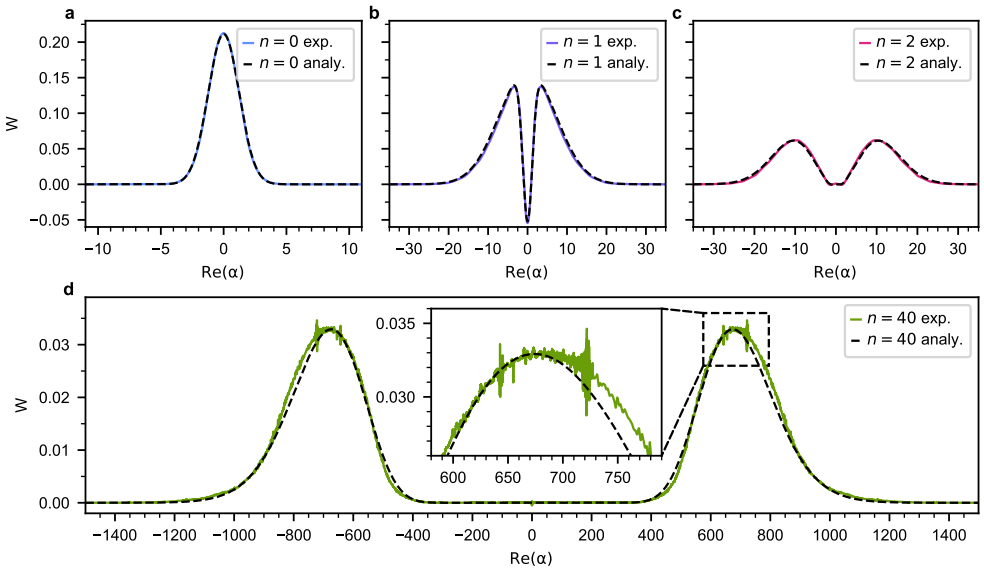

Fig. 3 Wigner functions of the reconstructed POVMs using the long-range smoothing (shown in Fig. 2(c)) and the analytical POVMs (shown in Fig. 2(d)) for a subset of outcomes (a) $n = 0$, (b) $n = 1$, (c) $n = 2$ and (d) $n = 40$. Clear negativity in (b) shows the non-classical nature of the corresponding POVM $\boldsymbol{\pi}_{n=1}$. The inset shows that the general overlap of the Wigner functions based on the experimental (colored lines) and analytical (black dashed lines) POVMs is good, however, some noise appears in the Wigner functions for the experimental POVMs at larger outcomes.

1.4 Application to other detector geometries

To compare with the current state of the art, we evaluate the proposed solution method and implementation for the detector geometry of Liu et al. [20]. They consider a spatially mutliplexed detector in the form of an on-chip network of beam splitters, whose output modes terminate in an SNSPD. This splits the input light equally onto many SNSPDs, thus gaining quasi-photon-number resolution. Their setting is characterized by the ratio of the maximal photon number in the Hilbert space and the number of outcomes, i.e., M/N being of the order of only ten. Specifically, the relation in their case is given in Eq. (5). The inputs for the minimisation problem, i.e., the matrices $\boldsymbol{F}$ and $\boldsymbol{P}$ were generated with the implementation of Liu et al. [31]. In our analysis of their data, we use a single compute node with two AMD EPYC 7763 64-core CPUs. Further software and hardware details can be found in Appendix A. The runtime and memory usage of the solver of Liu et al. [20] and the proposed method are shown in Fig. 4 for different N. While the CVXPY-MOSEK-based solver of Liu et al. has a lower runtime for small cases, i.e., $N \lesssim 30$, the solver proposed and implemented in this work is advantageous from intermediate sizes onwards. Performance of our proposed solver can be improved for small problems by using fewer CPU cores and, thus, threads which reduces the threading overhead. For $N = 1000$, which corresponds to $N \cdot M = 1.04 \cdot 10^7$ free variables in $\mathbf{\Pi}$, the estimated runtime and memory usage for

the CVXPY-MOSEK-based solver can be estimated to ≈ 75000 seconds and ≈ 12700 GiB while the proposed solver requires 1450 seconds and 1.6 GiB.

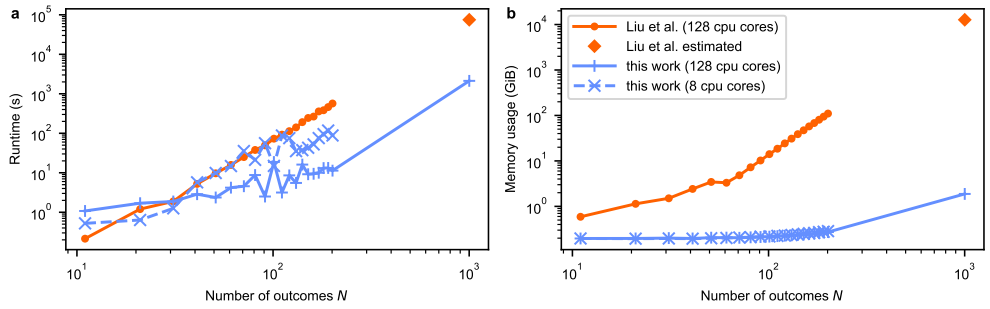

Fig. 4 Runtimes (a) and memory usage (b) of the CVXPY-MOSEK-solver of Liu et al. [20] (orange dots) and the solver proposed in this work (blue crosses, 8 CPU cores; blue pluses, 128 CPU cores) for the detector setting of Liu et al. for different numbers of outcomes N. The orange diamond symbol show the estimated runtime and memory usage based on linear extrapolation for the solver of Liu et al. for $N = 1000$.

1.5 Scalability for very large-scale problems

The limiting factor for the problem size is the main memory available on the compute nodes. We choose to evaluate the so-called "weak-scaling" scenario, in which the portion of the problem that one node works on is chosen to be constant. In this case, the maximum problem size depends on the available memory per compute node, the number of nodes, and their communication (message passing interface (MPI) ranks - see Methods, Memory usage, for details). Using our high-performance computing hardware, the scalability of the approach up to $3.4 \cdot 10^{12}$ free parameters has been demonstrated to be feasible, corresponding to about 27 TB of storage for the POVM matrix $\mathbf{\Pi}$.

The question naturally arises as to how large a system can be reasonably reconstructed using this method. This cannot be answered in general since the iterative numerical solution depends on the number of iterations required to converge, which in turn depends on the specific problem to be solved as well as the experimental input data. Thus, the scalability of the proposed solver and implementation is evaluated based on the scalability of the required performance-relevant operations (e.g. evaluation of the objective value, gradient, Hessian-matrix products, and others, as listed in Tab. 1 in the Methods) instead of the full reconstruction.

Nevertheless, one can estimate the scaling of the solver based on heuristics. Empirically for our two-stage iterative approach (see Methods, Algorithmic approach), the first stage requires about 10-15 Newton iterations, and the second stage requires about 30-200 Newton steps till convergence. Usually, between 20 and 50 conjugate-gradient iterations are needed per Newton iteration. Thus, the runtimes measured for individual operations (presented in Fig. 7 in the Methods, Scalability considerations) for

$N \cdot M$ up to $3 \cdot 10^{12}$ can be multiplied by lower and upper estimates of iterations to give lower and upper estimates for the runtime of the reconstruction.

Given these results, in Figure 5 we show estimated lower and upper limits of the runtime as applied to our detector geometry, up to a Hilbert space of $M \approx 3 \cdot 10^{12}$, number of outcomes $N \approx 150$ and requiring $D = \sqrt{M}$ input states to span the space. This is presented assuming that the number of iterations required for convergence remains similar to the experimentally realized case and that the number of compute nodes required to evaluate each system size is chosen according to the weak-scaling assumption (see Methods, Scalability considerations). The runtime for the reconstruction for the experimental setup of this work using one compute node of the Noctua 2 cluster [26] is also shown for comparison.

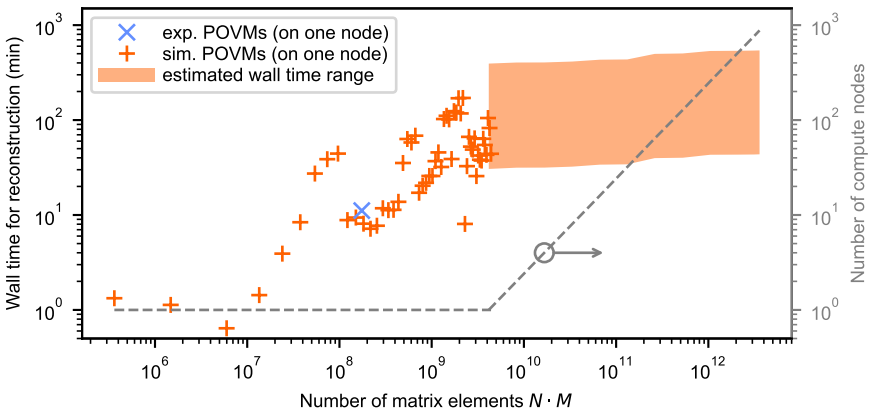

Fig. 5 Scaling of the reconstruction wall time for a simulated detector with $N = 151$ and $M = (D-1)^2$ up to one full compute node of the Noctua 2 cluster with 256 GiB of main memory ($N \cdot M \leq 4.4 \cdot 10^9$). Beyond $N \cdot M \approx 4.4 \cdot 10^9$ the wall time scaling for $N \approx 150$, $D = \sqrt{M} + 1$ is estimated from the measurements of the scalability of the underlying operations like the gradient combined with the estimated numbers of operations described in Sec. 1.5 assuming that the lowest possible number of compute nodes is used for the reconstruction, i.e., filling the main memory of the nodes. The blue cross represents the runtime for reconstructing the experimental POVMs using one compute node for comparison.

2 Discussion

Large-scale detectors for quantum light are becoming increasingly prevalent in quantum photonic technologies [10, 32, 33]. The ability to accurately characterise these devices, in a consistent quantum mechanical framework is central to using them effectively. This approach has been used to debug photonic quantum computing platforms [3], and could be applied to large-scale single-photon sensitive imaging systems [10, 22]. Developing computational tools to handle these large data sets, whilst preserving their quantum mechanical structure, is a vital task in the future proliferation of quantum photonic technologies.

We demonstrate the feasibility of applying high-performance computing to characterize a quantum detector that covers an extremely large outcome space, based on experimental data. The accuracy of this reconstruction is verified by comparison with an analytic model of the device, showing excellent agreement with fidelities above 99% per outcome. Moreover, we are able to plot Wigner functions of these very large quantum objects by exploiting arbitrary-precision floating-point numbers, and this freely available code is generally applicable for diagonal density matrices [34].

The solver itself, which is also freely available [34], is designed for use with phase-insensitive detectors. The degree and type of regularisation can be adapted to the specific detector type. Extensions of the solver to cover phase sensitivity are also discussed in the Methods section. Furthermore, the solver can be modified to solve other problems of the form of Eq. (3), provided the matrices fulfill the requirements (e.g. the banded structure of the $\boldsymbol{F}$-matrix) as outlined in the Methods section.

The main bottleneck of the solver is the available memory bandwidth. Thus, it might be worthwhile to consider and explore approximate computing techniques like reduced floating-point or fixed-point representations of numbers to further increase scale. In the other direction, using hardware with higher memory bandwidths like GPUs or FPGAs is a promising route.

3 Methods

3.1 Experiment

The experimental setup is shown in Fig. 1(a). We use a picosecond pulsed laser with a wavelength of 1556 nm and a repetition rate of 25 kHz to generate the coherent probe states. We can control the mean photon number per pulse by two variable optical attenuators that are placed after the laser. In order to ensure proper operation of the beam splitter loop (which is made from polarisation maintaining components) the polarisation is controlled by a manual fiber polarisation controller. Subsequently, the light pulses are coupled into the beam splitter loop with adaptable out-coupling R, loop-efficiency η_{loop} and temporal loop length τ. The light is partially coupled out of the loop into the time-bins of the time-multiplexed detector. These sub-pulses have a separation of $\tau = 156$ ns, which needs to be larger than the dead time of the SNSPD. Before the detector another polarisation controller is used in order to optimize the polarisation dependent detection efficiency of the SNSPD. Given the repetition rate of the laser and the bin separation τ a maximum of 256 time-bins are allowed. We use a total of $D = 1076$ different coherent probe states, whose mean photon numbers scale quadratically, in order to efficiently span the Hilbert space (with dimension $M \approx 1.2 \cdot 10^6$) of the detector. We note that the experiment was limited by the pulse energy of the laser and not the detector itself, as in principle it is not possible to saturate this type of multiplexed detector. We record raw time tags with a time tagger for $5 \cdot 10^5$ trials of every input state.

The coherent probe state matrix $\mathbf{F}$ is constructed by expanding the coherent states $d \in [0, D-1]$ in the photon-number basis according to Eq. (10) up to the Hilbert space dimension $i \in [0, M-1]$.

The outcome matrix $\mathbf{P}$ of the time-multiplexed detector is populated by counting the number of occupied time-bins per trial in a 5 ns coincidence window and dividing by the total number of trials. This leads to the probabilities $P_{d,n} = p_n|_d$ for different outcomes n and input states d. We can truncate the number of time-bins at 150 (resulting in $N = 151$ possible outcomes of the time-multiplexed detector), as subsequent time-bins are dominated by dark noise only.

As mentioned in Sec. 1.2 it is possible to derive the POVMs of this high dynamic range detector analytically. In order to calculate the analytical POVMs, we first need to find the experimental parameters of the device. This can be done by fitting the bin-click probabilities, i.e., the probabilities that a certain bin j fires for a given mean photon number. For a coherent state input to the detector, the bin-click probabilities are described by

$$p_j^{\mathrm{coh}}(d) = \begin{cases} 1 - \exp[-R\eta_{\mathrm{det}}|\alpha_d|^2] & j = 1 \\ 1 - \exp\left[-(1-R)^2 R^{-1} (R\eta_{\mathrm{loop}})^{j-1} |\alpha_d|^2 \eta_{\mathrm{det}}\right] & j \geq 2 \end{cases}, \tag{12}$$

which we adapted from Ref. [30] to include the detector efficiency η_{det}. We additionally neglect the dark-count probability, which is in the order of 5×10^{-8} per time-bin in a 5 ns coincidence window. We find the experimental parameters $R = 0.91644(9)$, $\eta_{\mathrm{loop}} = 0.90524(8)$ and $\eta_{\mathrm{det}} = 0.528(1)$ by fitting Eq. (12) simultaneously for all coherent input states $|\alpha_d|^2$ with $d \in [0, D-1]$.

Given the experimental parameters of the high dynamic range detector, we can then calculate the bin-click probabilities for photon-number (Fock) state inputs

$$p_j^{\mathrm{Fock}}(i) = \begin{cases} 1 - (1 - R\eta_{\mathrm{det}})^i & j = 1 \\ 1 - \left[1 - (1-R)^2 R^{-1} (R\eta_{\mathrm{loop}})^{j-1} \eta_{\mathrm{det}}\right]^i & j \geq 2 \end{cases}, \tag{13}$$

which is again adapted from Ref. [30] to include the detector efficiency η_{det}. With these bin-click probabilities, it is possible to use a closed-form expression for the Poisson binomial distribution [35] to calculate the POVMs of the detector (see Ref. [19] for more detail).

3.2 High-performance computing approach

3.2.1 Algorithmic approach

Instead of relying on general minimisation solvers such as CVXPY [24, 25] that are designed for handling arbitrary problems from large classes of minimisation problems, we propose a tailored algorithm that directly utilizes properties of the given problem.

We have evaluated several different approaches that only require the objective function and constraints as well as their derivatives but no matrix factorizations or decompositions. The goal of this restriction was to set up an algorithm that is well suited to be parallelised not only for using multiple CPU cores in one compute node but multiple CPU-nodes of a large HPC-cluster in parallel. Thus, we have evaluated

Table 1 The performance-relevant operations required for the approach described in section 3.2.1. $\boldsymbol{c}, \boldsymbol{d} \in \mathbb{R}^{M \times N}$, $\alpha \in \mathbb{R}$.

operation	equation	optimal parallelism
objective value	$O(\boldsymbol{\Pi}) = \|\|\boldsymbol{P} - \boldsymbol{F}\boldsymbol{\Pi}\|\|_2^2$	columns of $\boldsymbol{\Pi}$ with $D \times N$-reduction
gradient	$\partial_{\boldsymbol{\Pi}} O(\boldsymbol{\Pi})$	columns of $\boldsymbol{\Pi}$ with $D \times N$-reduction
Hessian products	$\boldsymbol{H}^{(k)}\boldsymbol{d}$	columns of $\boldsymbol{\Pi}$ with $D \times N$-reduction
diagonal of Hessian	$H^{(k)}_{i,j;i,j} = \frac{\partial^2 O(\Pi^{(k)})}{\partial \Pi_{i,j} \partial \Pi_{i,j}}$	columns of $\boldsymbol{\Pi}$ with $D \times N$-reduction
simplex-projection	$\mathcal{P}_{S^M}[\boldsymbol{d}]$	rows of $\boldsymbol{d}$
element-wise operations	e.g. $\boldsymbol{d} + \boldsymbol{c}, \boldsymbol{c} \cdot \boldsymbol{d}$	rows and/or columns of $\boldsymbol{d}$
scalar multiplications	$\alpha \boldsymbol{d}$	elements of $\boldsymbol{d}$
2-norm	$\sqrt{\sum_{i,n} d_{i,n}^2}$	rows/columns of $d_{i,j}$ with global reduction
row-maxima	$m(i) = \arg\max_{n \in \{0,\ldots,N-1\}} d_{i,n}$	rows of $\boldsymbol{d}$
row-sums	$s(i) = \arg \sum_{n=0}^{N-1} d_{i,n}$	rows of $\boldsymbol{d}$

the augmented Lagrangian approach [36, 37], different variants of active-space methods as well as projection approaches [38].

The minimisation problem at hand is characterized by a large number of equality (M) and inequality constraints ($N \cdot M$) and a relatively low computational complexity of the objective function due to the sparsity of $\boldsymbol{F}$. These characteristics directly impact the suitability of the classes of minimisation algorithms: For example, the large number of equality constraints cause a large number of additional augmentation and penalty terms in augmented Lagrangian-based approaches.

Our implemented approach [34] is based on the two-metric projected Newton method [27]. We developed a two-stage extension which showed significantly faster convergence. Both stages use a diagonally preconditioned conjugate gradient for the approximate solution of the linear system in Newton's method. As line search a backtracking approach with Armijo-like conditions [38] is used. The essential difference between the first stage and the second stage is that in the first stage a projection $\mathcal{P}_{\mathcal{S}^M}$ onto the M-dimensional unit simplex is used [39] whereas the second stage projects only onto the non-negativity constraints. Full details and algorithms are given in Appendix B. In the following section we present a few important aspects of the implementation.

3.2.2 Implementation

Required operations

The performance-relevant operations required for the solution approach described in Sec. 3.2.1 are listed in Table 1 together with their possible parallelism scope. The parallelism scope refers to the level or dimension along which the POVMs $\boldsymbol{\Pi}$ could be distributed to different processes or compute nodes without requiring communication or, if not possible, the additional required reduction is specified.

Data distribution and parallelisation

While all operations involving the objective function $O(\boldsymbol{\Pi}) = ||\boldsymbol{P} - \boldsymbol{F}\boldsymbol{\Pi}||_2^2$ have the columns of $\boldsymbol{\Pi}$ as a natural parallelism level due to the product $\boldsymbol{F}\boldsymbol{\Pi}$, all other required

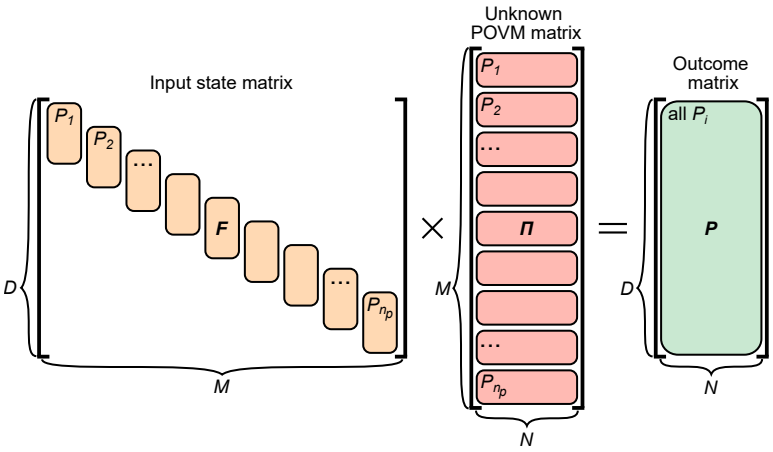

Fig. 6 Schematic representation of the distribution of $\boldsymbol{F}$, $\boldsymbol{\Pi}$, and $\boldsymbol{P}$ to the processes P_1,...,P_{n_p}. Colored areas show distribution units in parallelisation.

operations have at least the rows of $\boldsymbol{\Pi}$ as a natural parallelism level. Thankfully, for large M, the matrix $\boldsymbol{F}$ is sparse and banded, making it possible to also efficiently parallelise the operations that include the objective functions with respect to the rows of $\boldsymbol{\Pi}$. Thus, $\boldsymbol{\Pi}$ is distributed in blocks of rows to different processes. The matrices $\boldsymbol{P}$ and $\boldsymbol{P} - \boldsymbol{F\Pi}$ are replicated on every compute node. Only the relevant non-zero blocks of $\boldsymbol{F}$ are stored on the corresponding processes. The distribution schema is schematically shown in Fig. 6. Thus, the computation of $\boldsymbol{F\Pi}$ is performed in two steps: The first step is the process-local computation of the contributions to the auxiliary matrix $\boldsymbol{O} = \boldsymbol{F\Pi}$ with the rows of $\boldsymbol{\Pi}$ and the blocks of $\boldsymbol{F}$ that are present on the process. Secondly, the locally calculated contributions to $\boldsymbol{O}$ must be communicated to other processes. Due to the sparse banded structure of $\boldsymbol{F}$ for large M and because only parts of $\boldsymbol{O}$ are required on every process to compute the objective function, gradient, and Hessian products, no all-to-all communication is required for this step. Instead, the communication pattern can be handled efficiently with a butterfly graph. With the described distribution of blocks of rows of $\boldsymbol{\Pi}$, the proposed reconstruction can scale to very large problem sizes because the entire main memory available in a large HPC cluster can be used to compute a reconstruction.

We have implemented the distributed memory parallelisation with the message-passing interface (MPI) [40]. Additionally, we have parallelised the process-local operations with OpenMP [41] wherever possible. An emphasis was placed on avoiding false sharing of cache lines between the threads and an optimal usage of the available memory bandwidth.

3.2.3 Memory usage

The limiting factor for the size of the reconstruction in terms of the problem parameters N, M, and D that can be performed is the sum of the main memory available on the compute nodes.

The maximal possible Hilbert space dimension M_{max} possible for a reconstruction with our approach for a given number of outcomes N, number of probe states D, number of MPI-ranks n_{ranks} per compute node and number of compute nodes n_{nodes} and main memory per node mem_{node} can be estimated with

$$M_{\text{max}} \approx \frac{n_{\text{nodes}}}{6} \left(\frac{\text{mem}_{\text{node}}}{N \cdot 8 \text{ byte}} - 2DN_{\text{ranks}} \right). \tag{14}$$

3.2.4 Scalability considerations

Due to the dependence of the number of iterations during the minimisation on the input data, general statements on overall time-to-solution are limited. Nevertheless, the underlying operations can be analysed in detail. We have chosen the quadratic scaling of probe states, i.e., $M = (D-1)^2$ corresponding to the experimental situation presented in this work. The matrices $\boldsymbol{P}$ and the POVMs $\boldsymbol{\Pi}$ were chosen randomly and densely for the scalability experiments.

To analyse the scalability for very large reconstructions we chose a situation where the main memory of the available compute nodes is almost completely used and, thus, the limiting factor. Thus, the situation is related to the so-called weak-scaling case in parallelism where the portion of the problem that one compute node works on is kept constant when the number of compute nodes is increased.

For the compute nodes of Noctua 2 [26] with a main memory size of 256 GiB we have used $\text{mem}_{\text{node}} = 200$ GB to also account for buffer sizes of MPI transfers, OpenMP stack usage and other additional memory usages that are not covered in the approximation of Eq. (14). Scalability was investigated for 8 MPI ranks per node, i.e., one per NUMA domain, and 16 threads per MPI rank.

The scaling behavior of underlying operations is shown in Fig. 7 for the evaluation of the objective function (a), the Hessian-matrix product (b), gradient (c) and scalar products (d), which all require global communication between the compute nodes. Runtime results for the gradient are nearly indistinguishable from the Hessian-matrix product. Element-wise operations, scalar multiplications, and projections are trivially parallel with the proposed parallelisation scheme. For example, a conjugate-gradient iteration requires one Hessian product, several element-wise operations, and several scalar multiplications.

For larger photon number cutoffs $M \gtrsim 10^5$, the required runtime for the operations scales very favorably with the number of compute nodes. In contrast, for smaller M, the runtime increases visibly with the number of compute nodes. The underlying reason is that due to the choice of the weak-scaling scenario, in order to fill the main memory of the compute nodes in the case of small M, the number of outcomes N has to be very high as determined from Eq. (14).

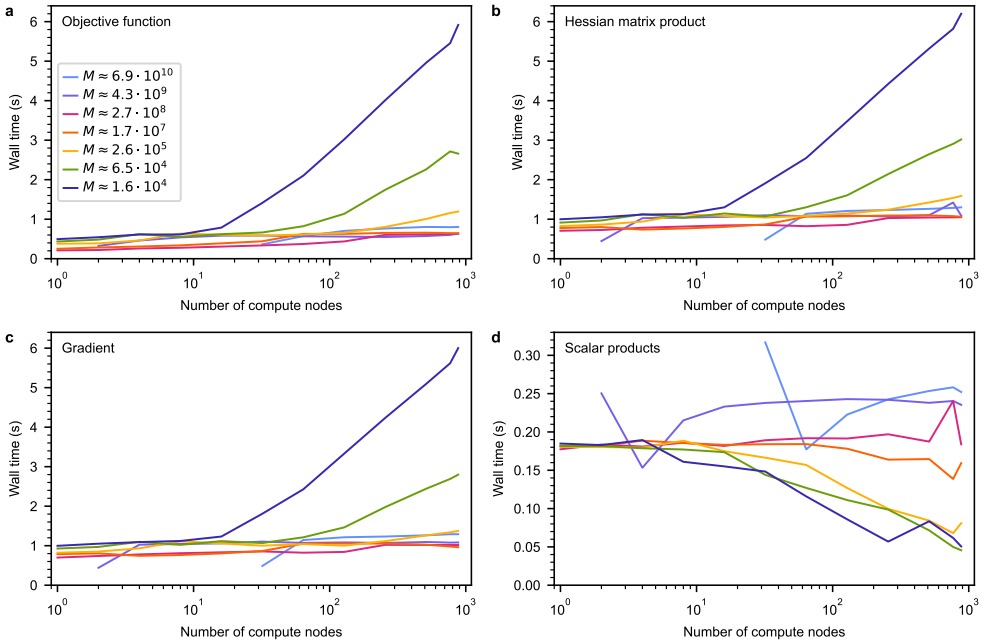

Fig. 7 Weak-scaling behavior of the runtime for the evaluation of the (a) objective function, (b) the Hessian-matrix product, (c) gradient and (d) scalar products.

Regularisation and smoothing

In addition to the least-squares objective function in Eq. (2a) our implementation also supports the next-neighbor regularisation term [7, 8]

$$g(\mathbf{\Pi}) = \gamma \sum_{n=0}^{N-1} \sum_{i=0}^{M-2} (\Pi_{i,n} - \Pi_{i+1,n})^2 \tag{15}$$

with the regularisation parameter $\gamma \geq 0$.

While long-range generalizations of the regularisation term are, in principle, easily possible, they introduce an ambiguity due to the choice of the averaging and the range-dependence of the weighting. The minimisation problem for this work's experimental situation exhibits a flat valley around the minimum. Thus, we propose a different way of encouraging a more physical, i.e., less noisy reconstruction by first performing a reconstruction starting with some initial $\mathbf{\Pi}$. We use $\Pi_{i,n}^{(0)} = 1/N$ as a starting point. The reconstruction problem is solved with the proposed two-stage algorithm and the result is smoothed by replacing each $\Pi_{i,n}$ with its long-range average

$$\tilde{\Pi}_{i,n} = \frac{1}{2N_{\text{smooth}}(i) + 1} \sum_{j=i-N_{\text{smooth}}(i)}^{i+N_{\text{smooth}}(i)} \Pi_{j,n}, \tag{16}$$

where $\mathbf{\Pi}$ is the result of the minimisation and the smoothing distance N_{smooth} depends on the photon number i so that the logarithmic photon-number scale in this experiment is accommodated. The smoothed POVMs $\tilde{\mathbf{\Pi}}$ then serve as the starting point for a second minimisation run where only the second stage is used.

The exclusion of the lowest $\sim$ 100 photon numbers from the smoothing step improves the results because it prevents a severe disruption of the qualitative structure at low photon numbers. We have found that a smoothing distance of $N_{\text{smooth}} = i/50$ yields a significant improvement over the POVMs without a smoothing step as shown in Fig. 2(c).

Possible improvements

While our implementation assumes no a priori sparsity pattern of $\mathbf{\Pi}$, $\boldsymbol{P}$, and $\boldsymbol{F}$, a sparsity pattern forms in $\mathbf{\Pi}$ during the minimisation. However, our current implementation performs all operations, like the gradients or Hessian-matrix-products, in the full $N \cdot M$ space, which reduces the code complexity but is likely wasteful in terms of memory bandwidth and floating-point operations. Also, due to the experimental setting with $M \gg N$, the parallelisation and handling of matrices, especially $\boldsymbol{F}$, are optimized for this situation.

3.2.5 Extension to phase-sensitive detectors

The algorithm proposed in the previous section is tailored for the detector tomography reconstruction problem of a phase-insensitive detector given in Eq. (3). For a phase-sensitive detector [8, 42, 43], the minimisation has the form

$$\min_{\boldsymbol{\pi}_n \in \mathbb{C}^{M\times M},\ n\in\{0,\dots,N-1\}} \quad ||\boldsymbol{P} - \boldsymbol{F}\mathbf{\Pi}||_2^2 \tag{17a}$$

$$\boldsymbol{\pi}_n = \boldsymbol{\pi}_n^\dagger \qquad \forall\ n \in \{0,...,N-1\} \tag{17b}$$

$$\boldsymbol{\pi}_n \succeq 0 \qquad \forall\ n \in \{0,...,N-1\} \tag{17c}$$

$$\textstyle\sum_{n=0}^{N-1} \boldsymbol{\pi}_n = \mathbb{1}, \tag{17d}$$

with $\mathbf{\Pi} = (\boldsymbol{\pi}_0, ..., \boldsymbol{\pi}_{N-1})$. The two-metric projected Newton approach can be generalized to this situation by replacing the projection on non-negativity constraints with a projection of a hermitian matrix on its nearest positive semi-definite matrix. Such a projection can, in the conceptually most straight-forward way, be calculated as

$$\mathcal{P}_{\text{semi-definite}}[\boldsymbol{A}] = \boldsymbol{U}^\dagger \begin{pmatrix} \max(0,\lambda_1) & & \\ & \max(0,\lambda_2) & \\ & & \dots \end{pmatrix} \boldsymbol{U}, \tag{18}$$

where the unitary matrix $\boldsymbol{U}$ contains the eigenvectors of $\boldsymbol{A}$ as columns and λ_i are the eigenvalues of $\boldsymbol{A}$. As this projection problem also arises in other areas, more efficient methods have been developed, especially for structured matrices [44, 45].

3.3 Computation of Wigner functions

The computation of Wigner functions for high photon numbers is numerically challenging due to the high absolute values occurring during the calculation. For the high photon numbers in this work, the intermediate values can easily exceed the maximal values that are representable with conventional double-precision floating-point numbers. Thus, we have set up a variant [34] by generalizing the implementation of the Wigner function available in QuantumOptics.jl [46] to use arbitrary-precision floating-point numbers via the BigFloat data type in Julia which uses the GNU MPFR library [47].

Data availability. The raw data, as well as the relevant matrices and Wigner function data is openly available via Zenodo at https://doi.org/10.5281/zenodo.10803758 and https://doi.org/10.5281/zenodo.10810148, respectively.

Code availability. The reconstruction code "Parallel Quantum Detector Tomography Solver" (pqdts [34]) and the code for calculating Wigner functions is openly available on GitHub via https://github.com/pc2/pqdts and can be cited via Zenodo https://zenodo.org/doi/10.5281/zenodo.10853650.

Acknowledgments. The authors thank Lorenzo M. Procopio and Thomas Hummel for valuable discussion. Funded by the European Union (ERC, QuESADILLA, 101042399). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This work has received funding from the German Ministry of Education and Research within the PhoQuant project (grant number 13N16103). The authors gratefully acknowledge the computing time provided to them on the high-performance computer Noctua 2 at the NHR Center PC2. These are funded by the Federal Ministry of Education and Research, and the state governments participate on the basis of the resolutions of the GWK for the national high-performance computing at universities (www.nhr-verein.de/unsere-partner).

Author contributions. T.S. and T.B. conceived the idea. T.S. designed the experiment, performed the measurements and analysed the measurement data. T.S. and M.L. performed pretesting for the large-scale reconstruction. R.S. programmed the solver, performed all benchmarking and calculated the Wigner functions. T.S., R.S. and T.B. wrote the manuscript with inputs from all authors. C.P. and T.B. supervised the project.

Competing interest. The authors declare no competing interests.

Supplementary information. Supplementary Information (see Appendix A and Appendix B) is provided alongside the main document, where software and hardware details can be found, as well as the complete algorithmic description and additional information for the Wigner function computation.

Appendix A Software and hardware details

All numerical results have been obtained using compute nodes of the Noctua 2 high-performance computing cluster [26]. Each node hosts two AMD EPYC 7763 CPUs with 64 CPU cores each and 256 GiB of DDR4-3200 main memory in an 8-channel-per-cpu configuration, giving a usable memory bandwidth of 370 GB/s in the STREAM Triad benchmark. The CPUs are configured with 4 NUMA domains per CPU, SMT is disabled, and turbo-mode is enabled. The following software versions were used: Python 3.11.5, CVXPY 1.4.2, and MOSEK 10.1.24 for the solver of Liu et al. and GCC 13.2.0 with OpenMPI 4.1.4 for the proposed solver.

Main memory usage refers to the maximal resident set size of the solver process as measured by the Linux kernel as the high-water mark (HWM).

Appendix B Algorithms

B.1 Two-metric projected Newton method

Bertsekas two-metric projected Newton method

The two-metric projected Newton method for a general constrained minimisation problem

$$\min_{\vec{x}\in\Omega} f(\vec{x}), \tag{B1}$$

where Ω is the space in which the constraints for $\vec{x}$ are fulfilled, is an iterative approach derived from the well-known Newton method for unconstrained minimisation [38]. For the unconstrained problem

$$\min_{\vec{x}\in\mathbb{R}^m} f(\vec{x}), \tag{B2}$$

the unconstrained Newton method approximates the objective function $f(\vec{x})$ in the k-th step by a quadratic model around the current iterate $\vec{x}^{(k)}$ as

$$Q^{(k)}(\vec{x}) \approx f(\vec{x}^{(k)}) + (\vec{x}-\vec{x}^{(k)})^T \nabla f(\vec{x}^{(k)}) + \tfrac{1}{2}(\vec{x}-\vec{x}^{(k)})^T \boldsymbol{H}^{(k)} (\vec{x}-\vec{x}^{(k)}), \tag{B3}$$

where $\boldsymbol{H}^{(k)}$ is the Hessian matrix in the k-th step, i.e. $H_{ij}^{(k)} = \frac{\partial^2}{\partial x_i \partial x_j} f(\vec{x}^{(k)})$. The next iterate $\vec{x}^{(k+1)}$ is then the solution of the minimisation problem:

$$\vec{x}^{(k+1)} = \min_{\vec{x}\in\mathbb{R}^m} Q^{(k)}(\vec{x}). \tag{B4}$$

The quadratic unconstrained minimisation problem in Eq. (B4) can be rewritten as the well-known linear system for the minimisation step $\vec{p}^{(k)} = \vec{x}^{(k+1)} - \vec{x}^{(k)}$:

$$\boldsymbol{H}^{(k)} \vec{p}^{(k)} = -\nabla f(\vec{x}^{(k)}), \tag{B5}$$

which can be solved with various numerical methods. Usually, this schema is augmented by a line-search procedure where the next iterate $\vec{x}^{(k+1)}$ is determined from the step vector $\vec{p}^{(k)}$ by the solution of the line-search problem

$$\alpha^{(k)} = \arg\min_{\alpha \in \mathbb{R}} f(\vec{x}^{(k)} + \alpha \vec{p}^{(k)}) \tag{B6}$$

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} + \alpha^{(k)} \vec{p}^{(k)}. \tag{B7}$$

While this additional line search procedure only adds a typically negligible computational cost, the approach is much more numerically reliable. For an objective function with positive definite Hessian, the unconstrained Newton method converges quadratically fast near the solution [38].

Analogous to the translation of the gradient descent method for unconstrained problems to the gradient projection method for constrained problems, the Newton method for unconstrained problems, Eq. (B3)-(B7) can be translated to problems with constraints [38]. For this purpose, the unconstrained quadratic approximation in Eq. (B3) is written as a constrained problem that determines the next iterate $\vec{x}^{(k+1)}$ as

$$\vec{x}^{(k+1)} = \arg\min_{\vec{x} \in \Omega} Q^{(k)}(\vec{x}). \tag{B8}$$

the crucial difference is that the minimisation, in contrast to the unconstrained case in Eq. (B4), is restricted to vectors $\vec{x}$ that fulfill the constraints. Consequently, the step direction $\vec{p}^{(k)}$ cannot be obtained by the solution of the linear system Eq. (B5), and instead, the constrained problem of Eq. (B8) is typically significantly harder to solve. In the case of bound-constrained problems, i.e. $\Omega = \{\vec{x} \in \mathbb{R}^m : l \leq x_i \leq u \ \forall i\}$, one possibility to circumvent this issue is the idea of two-metric projected Newton approaches [27]. Equation (B8) is rewritten as a projection

$$\vec{x}^{(k+1)} = \arg\min_{\vec{x} \in \mathbb{R}^m : l \leq x_i \leq u} \frac{1}{2} ||\vec{x} - (\vec{x}^{(k)} - (\boldsymbol{H}^{(k)})^{-1} \nabla f(\vec{x}^{(k)}))||^2_{\boldsymbol{H}^{(k)}}, \tag{B9}$$

where the norm with respect to the metric defined by the positive-definite matrix $\boldsymbol{A}$ is defined as $||\vec{y}||_{\boldsymbol{A}} = \sqrt{\vec{y}^T \boldsymbol{A} \vec{y}}$. Instead of solving this projection with respect to the metric induced by the Hessian matrix $\boldsymbol{H}$, the two-metric approach solves the projection problem

$$\vec{x}^{(k+1)} \approx \arg\min_{\vec{x} \in \mathbb{R}^m : l \leq x_i \leq u} \frac{1}{2} ||\vec{x} - (\vec{x}^{(k)} - (\boldsymbol{D}^{(k)})^{-1} \nabla f(\vec{x}^{(k)}))||^2_{\boldsymbol{I}} \tag{B10}$$

with respect to the unit metric, but the inverse of the Hessian matrix is replaced by the inverse of a suitable positive definite matrix $\boldsymbol{D}$. Practically, the projection with respect to the unit metric in Eq. (B10) has the solution

$$\vec{x}^{(k+1)} \approx \mathcal{P}_{u,l}[\vec{x}^{(k)} - (\boldsymbol{D}^{(k)})^{-1} \nabla f(\vec{x}^{(k)})], \tag{B11}$$

where $\mathcal{P}_{u,l}(\vec{x})$ is the projection of the vector $\vec{x}$ onto the constraint surface, i.e.,

$$(\mathcal{P}_{u,l}(\vec{x}))_i = \begin{cases} l, \text{if } x_i < l \\ u, \text{if } x_i > u \\ x_i, \text{otherwise} \end{cases} . \tag{B12}$$

A line-search method is added to this approach as

$$\alpha^{(k)} = \arg\min_{\alpha \in \mathbb{R}} \mathcal{P}_{u,l}[\vec{x}^{(k)} - \alpha (\boldsymbol{D}^{(k)})^{-1} \nabla f(\vec{x}^{(k)})] \tag{B13}$$

$$\vec{x}^{(k+1)} = \mathcal{P}_{u,l}[\vec{x}^{(k)} - \alpha^{(k)} (\boldsymbol{D}^{(k)})^{-1} \nabla f(\vec{x}^{(k)})] \tag{B14}$$

Bertsekas [27] has shown that for a problem with non-negativity constraints,

$$\min_{\vec{x} \in \mathbb{R}^m : x_i \geq 0 \ \forall i} f(\vec{x}) \tag{B15}$$

a choice of the matrix $\boldsymbol{D}^{(k)}$ as

$$(D^{(k)})_{ij} = \begin{cases} 0, \text{ if } i \neq j \text{ and either } i \in I \text{ or } j \in I \\ H_{ij}^{(k)}, \text{ otherwise} \end{cases} \tag{B16}$$

with the index set I defined as

$$I = \{i : 0 \leq x_i \leq \epsilon, \tfrac{\partial f(\vec{x})}{\partial x_i} > 0\} \tag{B17}$$

and a small ϵ, this approach leads to a globally convergent algorithm with a superlinear convergence rate under mild conditions.

For minimisation problems over a probability simplex,

$$\min_{\vec{x} \in \mathbb{R}^m : x_i \geq 0 \ \forall i, \ \sum_{i=1}^m x_i = 1} f(\vec{x}), \tag{B18}$$

Bertsekas [27] has proposed an approach that transforms the problem so that it can be handled with the two-metric projected Newton approach defined in Eq. (B14)-(B17). The transformation can be generalized straightforwardly to the tomography reconstruction problem in Eq. (3), which has M sum-constraints. The approach only requires evaluations of the objective function, the gradient, and the approximate solution of the linear system

$$\boldsymbol{D}^{(k)} \vec{p}^{(k)} = -\nabla f(\vec{x}^{(k)}) \tag{B19}$$

for $\vec{p}^{(k)}$, which can be implemented as an iterative preconditioned conjugate-gradient (CG) method where only matrix-product operations between $\boldsymbol{D}^{(k)}$ and vectors as well as vector-vector operations are required. Thus, the memory usage footprint is only a few times larger than the memory usage for the storage of the $\boldsymbol{P}$, $\boldsymbol{F}$, and $\boldsymbol{\Pi}$ given

in Eq. (7). The complete two-metric truncated Newton algorithm for the detector tomography problem Eq. (3) is given in Algo. 1.

Algorithm 1 Two-metric projected truncated Newton algorithm

$\Pi_{i,n}^{(0)} \leftarrow \frac{1}{N}\ \forall i,n$ ▷ Initialization that fulfills the constraints
choose $0 < \beta < 1$
for $k \leftarrow 0,1,2,...$ **do** ▷ projected Newton iteration
  for $i \leftarrow 0,1,2,...,M$ **do** ▷ transformation step
    $m(i) \leftarrow \arg\max_{n\in\{0,...,N-1\}} \Pi_{i,n}^{(k)}$ ▷ determine per-row maxima
    $\tilde{\Pi}_{i,n}^{(k)} \leftarrow \Pi_{i,n}^{(k)}\ \forall n \neq m(i)$
    $\tilde{\Pi}_{i,m(i)}^{(k)} \leftarrow 1 - \sum_{j\neq m(i)} \Pi_{i,j}^{(k)}$ ▷ implicitly account for sum-constraint
  end for
  $\tilde{\boldsymbol{G}}^{(k)} \leftarrow -\partial_{\tilde{\boldsymbol{\Pi}}^{(k)}} ||\boldsymbol{P} - \boldsymbol{F}\boldsymbol{\Pi}^{(k)}||_2^2$ ▷ gradient with respect to $\tilde{\boldsymbol{\Pi}}$
  solve $\boldsymbol{D}^{(k)}\boldsymbol{P}^{(k)} = \tilde{\boldsymbol{G}}^{(k)}$ ▷ with diagonally-preconditioned CG
  for $l \leftarrow 0,1,2,...$ **do** ▷ line search
    $\alpha^{(k)} \leftarrow \beta^l$
    $\Pi_{i,n}^{(k+1)} \leftarrow \mathcal{P}_{0,\infty}[\tilde{\boldsymbol{\Pi}}^{(k)} + \alpha^{(k)}\boldsymbol{P}^{(k)}]_{i,n}\ \forall i, n \neq m(i)$ ▷ update $\boldsymbol{\Pi}$
    $\Pi_{i,m(i)}^{(k+1)} \leftarrow 1 - \sum_{j\neq m(i)} \Pi_{i,j}^{(k+1)}$
    if $\Pi_{i,m(i)}^{(k+1)} \geq 0\ \forall i$ and $\boldsymbol{\Pi}^{(k+1)}$ fulfills Armijo-conditions [38] **then**
      exit loop
    end if
  end for
  if converged **then**
    exit loop
  end if
end for

Two-stage extension to the two-metric projected Newton approach

While the algorithm described in the previous section has a favorable memory usage characteristic and can be parallelised efficiently, we have found that due to a large number of sum constraints, Eq. (3c), the method converges slowly if it is not already rather close to the solution. The underlying reason is that in the line search in Algorithm 1 the elements of $\boldsymbol{\Pi}^{(k+1)}$ corresponding to the row-wise maxima are obtained implicitly from the sum-constraint as

$$\Pi_{i,m(i)}^{(k+1)} = 1 - \sum_{j\neq m(i)} \Pi_{i,j}^{(k+1)}, \tag{B20}$$

where $m(i)$ is the column-index of the maximum in the row i of the matrix $\boldsymbol{\Pi}^{(k)}$. If $\boldsymbol{\Pi}^{(k)}$ is not close to the solution of the minimisation problem, there is likely a row in

$\mathbf{\Pi}^{(k)}$ for which $\Pi^{(k)}_{i,m(i)} \ll 1$ and which, thus, forces the step length $\alpha^{(k)}$ to be small to fulfill the non-negativity constraint for $\Pi^{(k+1)}_{i,m(i)}$. Thus, in the next section, we propose a two-stage approach that avoids this convergence slowdown. In the first stage, we use a modified variant of the two-metric projected Newton method of Bertsekas by replacing the transformation step for the sum constraints with a projection onto the probability simplex that also enforces the sum constraints. The projection $\mathcal{P}_S[\vec{x}]$ of a vector $\vec{x} \in \mathbb{R}^m$ onto the probability simplex $S = \{\vec{y} \in \mathbb{R}^m | y_i \geq 0, \sum_i y_i = 1\}$ is defined as the point closest to $\vec{x}$ that is on S, i.e.,

$$\mathcal{P}_S[\vec{x}] = \arg\min_{\vec{y} \in S} ||\vec{x} - \vec{y}||_2. \tag{B21}$$

In practice, we use the algorithm proposed by Condat [39] that has a best-case complexity of $\mathcal{O}(m)$, a worst-case complexity of $\mathcal{O}(m^2)$, and additional memory requirement $\mathcal{O}(1)$. For the purpose of the detector tomography reconstruction problem in Eq. (3), an M-dimensional generalization $\mathcal{P}_{S^M}$ of projection on a probability simplex can be defined as

$$\mathcal{P}_{S^M}(\mathbf{\Pi}) = \begin{pmatrix} \mathcal{P}_S[(\Pi_{0,0}, ..., \Pi_{0,N-1})] \\ \vdots \\ \mathcal{P}_S[(\Pi_{M-1,0}, ..., \Pi_{M-1,N-1})] \end{pmatrix} \tag{B22}$$

that projects the POVM $\mathbf{\Pi}$ on the constraints. The resulting algorithm is shown in Algo. 2. However, due to the presence of the M-dimensional projection $\mathcal{P}_{S^M}$, we have not yet been able to formally show global convergence for this modified variant and, thus, only use it as a first stage to accelerate convergence towards the solution. Once sufficiently close to the solution, we switch to the second stage with the proven globally convergent two-metric projected Newton approach in Algo. 1. Practical results that demonstrate the efficiency of this approach are shown in Fig. B1. The overall two-stage, two-metric projected Newton approach is shown in Algo. 3. Thus, the two-stage algorithm is globally convergent and improves on the convergence issue of Algo. 1. The only required expensive operations are evaluating the objective function, gradient, and products of the modified Hessian $\boldsymbol{D}^{(k)}$ with vectors for the conjugate-gradient procedure. The constraints are enforced to numerical precision in both stages. The memory usage of the algorithm is given as the sum of the memory needed to store the matrices $\boldsymbol{F}$, $\boldsymbol{P}$, $\mathbf{\Pi}^{(k)}$ plus the auxiliary matrices $\boldsymbol{G}^{(k)}$, $\boldsymbol{O}^{(k)} = \boldsymbol{F}\mathbf{\Pi}^{(k)}$, $\mathbf{\Pi}^{(k+1)}$, the index array I, and auxiliary matrices for the diagonally-preconditioned CG iteration. The overall memory requirement is

$$\text{mem}_{\text{2metric}} = (2ND + 6NM + MD + \mathcal{O}(M) + \mathcal{O}(N)) \cdot 8 \text{ byte}. \tag{B23}$$

For large M, a sparse storage of $\boldsymbol{F}$ instead of a dense representation is used to drastically reduce the term MD in the memory estimation. For comparison, in the case of the detector geometry of Liu et al., i.e., the dependence of M and D given by

Eq. (5)-(6), and sufficiently large N, the memory estimate given in Eq. (B23) results in

$$\text{mem}_{\text{2metric,Liu}} \lesssim 6.8 \cdot 10^{-7} \cdot N^{2.2} \text{ GiB}. \tag{B24}$$

Algorithm 2 Two-metric projected truncated Newton algorithm

$\Pi_{i,n}^{(0)} \leftarrow \frac{1}{N} \; \forall i, n$ ▷ Initialization that fulfills the constraints
choose $0 < \beta < 1$
for $k \leftarrow 0, 1, 2, \ldots$ **do** ▷ projected Newton iteration
  $\boldsymbol{G}^{(k)} \leftarrow -\partial_{\boldsymbol{\Pi}^{(k)}} ||\boldsymbol{P} - \boldsymbol{F}\boldsymbol{\Pi}^{(k)}||_2^2$ ▷ gradient with respect to $\boldsymbol{\Pi}$
  solve $\boldsymbol{D}^{(k)}\boldsymbol{P}^{(k)} = \boldsymbol{G}^{(k)}$ ▷ with diagonally-preconditioned CG
  for $l \leftarrow 0, 1, 2, \ldots$ **do** ▷ line search
    $\alpha^{(k)} \leftarrow \beta^l$
    $\boldsymbol{\Pi}^{(k+1)} \leftarrow \mathcal{P}_{S^{M+1}}[\boldsymbol{\Pi}^{(k)} + \alpha^{(k)}\boldsymbol{P}^{(k)}]$ ▷ $M+1$-dimensional projection
    if $\boldsymbol{\Pi}^{(k+1)}$ fulfills Armijo-conditions **then**
      exit loop
    end if
  end for
  if converged **then**
    exit loop
  end if
end for

Algorithm 3 Two-stage, two-metric projected truncated Newton algorithm

$\Pi_{i,n}^{(0)} \leftarrow \frac{1}{N} \; \forall i, n$ ▷ Initialization that fulfills the constraints
choose $0 < \beta < 1$
minimize $\boldsymbol{\Pi}$ with Algo. 2 starting from $\boldsymbol{\Pi}^{(0)}$ ▷ Stage 1
minimize $\boldsymbol{\Pi}$ with Algo. 1 starting from the result of the previous stage ▷ Stage 2

The convergence criterion for the line searches in both stages are the Armijo-like conditions where the smallest $m \in \mathbb{N}_0$ that fulfills

$$f(\mathcal{P}[\boldsymbol{\Pi}^{(k)} + \beta^m \boldsymbol{P}^{(k)}]) \leq f(\boldsymbol{\Pi}^{(k)}) + c\beta^m \frac{\partial}{\partial \alpha} f(\mathcal{P}[\boldsymbol{\Pi}^{(k)} + \alpha \boldsymbol{P}^{(k)}])|_{\alpha=0} \tag{B25}$$

with the objective function f, $\beta = \frac{3}{4}$, and $c = \frac{1}{10}$ is used for the step length $\alpha^{(k)} = \beta^m$ [27]. For the second stage also the non-negativity conditions, $\Pi_{i,n} \geq 0$, have to be satisfied explicitly which is fulfilled implicitly for the first stage. For the convergence criterion for the transition from stage 1 to stage 2, the criterion

$$|\partial_\alpha f(\mathcal{P}_{S^M}[\boldsymbol{\Pi}^{(k)} + \alpha \boldsymbol{P}^{(k)}])| \leq 10^{-4} \tag{B26}$$

is chosen. The convergence criterion for the second stage and, thus, for the overall algorithm is derived from the KKT-conditions [38] of the minimisation problem and defined as

$$\sqrt{\frac{1}{NM}\sum_{n=0}^{N-1}\sum_{i=0}^{M-1}(\Pi_{i,n}\cdot(\partial_{\Pi_{i,n}}f(\mathbf{\Pi})+\lambda_i))^2} \leq \epsilon_{\mathrm{KKT}}, \tag{B27}$$

where $\lambda_i = \max(0, -\min_n \partial_{\Pi_{i,n}} f(\mathbf{\Pi}))$ is the estimate for the Lagrange multipliers of the sum-constraints.

Convergence of the two-stage approach

We investigate the convergence behavior of the proposed method with the experimental measurement data, i.e., with a Hilbert space cutoff of $M = 1210581$, $N = 151$ outcomes, and $D = 1076$ probe states.

The effectiveness of the proposed two-stage approach can be seen by comparing the convergence speed when only using the second stage, i.e., Bertseka's two-metric projected Newton method in contrast to the two-stage approach depicted in Fig. B1. While the convergence close to the minimum is very similar by construction, replacing the first iterations with the modified variant drastically accelerates convergence when far away from the minimum.

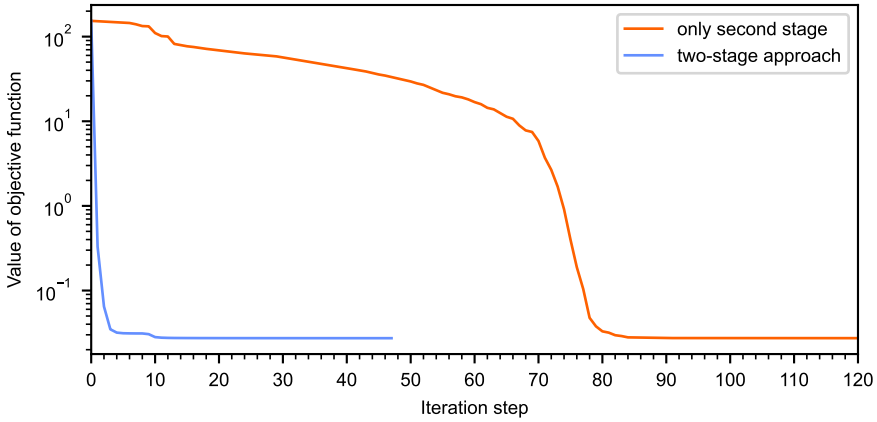

Fig. B1 Comparison of the convergence of the proposed two-stage approach with the convergence when only using the second stage. No regularisation was applied, i.e., $\gamma = 0$.

B.2 Wigner function

Computations with arbitrary-precision floating-point numbers required for the computation of the Wigner function in Sec. 3.3 are much more expensive than double-precision floating-point numbers because no direct hardware implementation is

available. We have optimized the implementation of the Wigner function for phase-insensitive detectors, i.e., diagonal density matrices, and have implemented a trivial parallelisation via MPI to scale the computation of the Wigner function to many CPU cores/compute nodes [34].

The arbitrary-precision floating-point numbers in MPFR are represented as the product of a sign $s \in \{-1, 1\}$, a 64-bit exponent e, and an arbitrary-sized fixed-point mantissa m as

$$s \cdot 2^e \cdot m. \tag{B28}$$

The important ingredient is the very large range $\approx 10^{2^{62}} \approx 10^{10^{18}}$ of numbers that can be represented compared to the range of 64-bit floating-point numbers with a range of $\approx 10^{307}$. The bit size of the mantissa can be varied. For the Wigner functions in this work, mantissa sizes of about 60-70 bits are sufficient in the sense that larger mantissa sizes yield binary-identical Wigner functions when the final result is converted to double-precision floating-point numbers.

References

[1] Aspuru-Guzik, A., Walther, P.: Photonic quantum simulators. Nature Physics **8**(4), 285–291 (2012) https://doi.org/10.1038/nphys2253

[2] Aaronson, S., Arkhipov, A.: The computational complexity of linear optics. In: Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing. STOC '11, pp. 333–342. ACM, New York, NY, USA (2011). https://doi.org/10.1145/1993636.1993682

[3] Deng, Y.-H., Gu, Y.-C., Liu, H.-L., Gong, S.-Q., Su, H., Zhang, Z.-J., Tang, H.-Y., Jia, M.-H., Xu, J.-M., Chen, M.-C., Qin, J., Peng, L.-C., Yan, J., Hu, Y., Huang, J., Li, H., Li, Y., Chen, Y., Jiang, X., Gan, L., Yang, G., You, L., Li, L., Zhong, H.-S., Wang, H., Liu, N.-L., Renema, J.J., Lu, C.-Y., Pan, J.-W.: Gaussian Boson Sampling with Pseudo-Photon-Number-Resolving Detectors and Quantum Computational Advantage. Physical Review Letters **131**(15), 150601 (2023) https://doi.org/10.1103/PhysRevLett.131.150601

[4] Luis, A., Sánchez-Soto, L.L.: Complete Characterization of Arbitrary Quantum Measurement Processes. Physical Review Letters **83**(18), 3573–3576 (1999) https://doi.org/10.1103/PhysRevLett.83.3573

[5] Fiurášek, J.: Maximum-likelihood estimation of quantum measurement. Physical Review A **64**(2), 24102 (2001) https://doi.org/10.1103/PhysRevA.64.024102

[6] D'Ariano, G.M., Maccone, L., Presti, P.L.: Quantum Calibration of Measurement Instrumentation. Physical Review Letters **93**(25), 250407 (2004) https://doi.org/10.1103/PhysRevLett.93.250407

[7] Lundeen, J.S., Feito, A., Coldenstrodt-Ronge, H., Pregnell, K.L., Silberhorn, C., Ralph, T.C., Eisert, J., Plenio, M.B., Walmsley, I.A.: Tomography of quantum

detectors. Nature Physics **5**(1), 27–30 (2009) https://doi.org/10.1038/nphys1133

[8] Feito, A., Lundeen, J.S., Coldenstrodt-Ronge, H., Eisert, J., Plenio, M.B., Walmsley, I.A.: Measuring measurement: theory and practice. New Journal of Physics **11**(9), 93038 (2009) https://doi.org/10.1088/1367-2630/11/9/093038

[9] Coldenstrodt-Ronge, H.B., Lundeen, J.S., Pregnell, K.L., Feito, A., Smith, B.J., Mauerer, W., Silberhorn, C., Eisert, J., Plenio, M.B., Walmsley, I.A.: A proposed testbed for detector tomography. Journal of Modern Optics **56**(2-3), 432–441 (2009) https://doi.org/10.1080/09500340802304929

[10] Oripov, B.G., Rampini, D.S., Allmaras, J., Shaw, M.D., Nam, S.W., Korzh, B., McCaughan, A.N.: A superconducting nanowire single-photon camera with 400,000 pixels. Nature **622**(7984), 730–734 (2023) https://doi.org/10.1038/s41586-023-06550-2

[11] Brida, G., Ciavarella, L., Degiovanni, I.P., Genovese, M., Lolli, L., Mingolla, M.G., Piacentini, F., Rajteri, M., Taralli, E., Paris, M.G.A.: Quantum characterization of superconducting photon counters. New Journal of Physics **14**(8), 85001 (2012) https://doi.org/10.1088/1367-2630/14/8/085001

[12] Humphreys, P.C., Metcalf, B.J., Gerrits, T., Hiemstra, T., Lita, A.E., Nunn, J., Nam, S.W., Datta, A., Kolthammer, W.S., Walmsley, I.A.: Tomography of photon-number resolving continuous-output detectors. New Journal of Physics **17**(10), 103044 (2015) https://doi.org/10.1088/1367-2630/17/10/103044

[13] Schapeler, T., Philipp Höpker, J., Bartley, T.J.: Quantum detector tomography of a 2×2 multi-pixel array of superconducting nanowire single photon detectors. Optics Express **28**(22), 33035–33043 (2020) https://doi.org/10.1364/OE.404285

[14] Endo, M., Sonoyama, T., Matsuyama, M., Okamoto, F., Miki, S., Yabuno, M., China, F., Terai, H., Furusawa, A.: Quantum detector tomography of a superconducting nanostrip photon-number-resolving detector. Optics Express **29**(8), 11728 (2021) https://doi.org/10.1364/OE.423142 2102.09712

[15] Cai, Y., Chen, Y., Chen, X., Wu, G., Wu, E.: Quantum characteristics and applications of multi-pixel photon counter. Microwave and Optical Technology Letters **63**(8), 2052–2057 (2021) https://doi.org/10.1002/mop.32865

[16] Fitzke, E., Krebs, R., Haase, T., Mengler, M., Alber, G., Walther, T.: Time-dependent POVM reconstruction for single-photon avalanche photo diodes using adaptive regularization. New Journal of Physics (2022) https://doi.org/10.1088/1367-2630/ac5004

[17] Santana, T., Muñoz, C., Chunnilall, C.: Extending the quantum tomography of a quasi-photon-number-resolving detector (2023) https://doi.org/10.1364/opticaopen.24908667.v1

[18] Cooper, M., Karpiński, M., Smith, B.J.: Local mapping of detector response for reliable quantum state estimation. Nature Communications **5**(1), 4332 (2014) https://doi.org/10.1038/ncomms5332

[19] Schapeler, T., Höpker, J.P., Bartley, T.J.: Quantum detector tomography of a high dynamic-range superconducting nanowire single-photon detector. Superconductor Science and Technology **34**(6), 64002 (2021) https://doi.org/10.1088/1361-6668/abee9a

[20] Liu, D.-S., Wang, J.-Q., Zou, C.-L., Ren, X.-F., Guo, G.-C.: Optimized detector tomography for photon-number-resolving detectors with hundreds of pixels. Physical Review A **108**(5), 052611 (2023) https://doi.org/10.1103/PhysRevA.108.052611

[21] Zhang, L., Coldenstrodt-Ronge, H.B., Datta, A., Puentes, G., Lundeen, J.S., Jin, X.-M., Smith, B.J., Plenio, M.B., Walmsley, I.A.: Mapping coherence in measurement via full quantum tomography of a hybrid optical detector. Nature Photonics **6**(6), 364–368 (2012) https://doi.org/10.1038/nphoton.2012.107

[22] Morimoto, K., Ardelean, A., Wu, M.-L., Ulku, A.C., Antolovic, I.M., Bruschini, C., Charbon, E.: Megapixel time-gated SPAD image sensor for 2D and 3D imaging applications. Optica **7**(4), 346 (2020) https://doi.org/10.1364/OPTICA.386574

[23] MOSEK ApS: The MOSEK Optimization Toolbox for MATLAB Manual. Version 10.1. (2023). http://docs.mosek.com/10.1/toolbox/index.html

[24] Diamond, S., Boyd, S.: CVXPY: A Python-embedded modeling language for convex optimization. Journal of Machine Learning Research **17**(83), 1–5 (2016)

[25] Agrawal, A., Verschueren, R., Diamond, S., Boyd, S.: A rewriting system for convex optimization problems. Journal of Control and Decision **5**(1), 42–60 (2018) https://doi.org/10.1080/23307706.2017.1397554

[26] Bauer, C., Kenter, T., Lass, M., Mazur, L., Meyer, M., Nitsche, H., Riebler, H., Schade, R., Schwarz, M., Winnwa, N., Wiens, A., Wu, X., Plessl, C., Simon, J.: Noctua 2 supercomputer. Journal of large-scale research facilities JLSRF (2024). In press.

[27] Bertsekas, D.P.: Projected newton methods for optimization problems with simple constraints. SIAM Journal on Control and Optimization **20**(2), 221–246 (1982) https://doi.org/10.1137/0320018 https://doi.org/10.1137/0320018

[28] Landi, G., Loli Piccolomini, E.: A projected Newton-CG method for nonnegative astronomical image deblurring. Numerical Algorithms **48**(4), 279–300 (2008) https://doi.org/10.1007/s11075-008-9198-3

[29] Schmidt, M., Kim, D., Sra, S.: Projected Newton-type Methods in Machine

Learning. The MIT Press (2011). https://doi.org/10.7551/mitpress/8996.003.0013

[30] Tiedau, J., Meyer-Scott, E., Nitsche, T., Barkhofen, S., Bartley, T.J., Silberhorn, C.: A high dynamic range optical detector for measuring single photons and bright light. Optics Express **27**(1), 1–15 (2019) https://doi.org/10.1364/OE.27.000001

[31] Liu, D.-S., Wang, J.-Q., Zou, C.-L., Ren, X.-F., Guo, G.-C.: Optimized detector tomography for photon-number resolving detectors with hundreds of pixels (2023). https://github.com/DS-Liu/Modified-detector-tomography

[32] Cheng, R., Zhou, Y., Wang, S., Shen, M., Taher, T., Tang, H.X.: A 100-pixel photon-number-resolving detector unveiling photon statistics. Nature Photonics **17**(1), 112–119 (2023) https://doi.org/10.1038/s41566-022-01119-3

[33] Eaton, M., Hossameldin, A., Birrittella, R.J., Alsing, P.M., Gerry, C.C., Dong, H., Cuevas, C., Pfister, O.: Resolution of 100 photons and quantum generation of unbiased random numbers. Nature Photonics **17**(1), 106–111 (2023) https://doi.org/10.1038/s41566-022-01105-9

[34] Schade, R., Lass, M., Schapeler, T., Plessl, C., Bartley, T.J.: Parallel Quantum Detector Tomography Solver (pqdts). https://doi.org/10.5281/zenodo.10908474 . https://github.com/pc2/pqdts

[35] Fernandez, M., Williams, S.: Closed-Form Expression for the Poisson-Binomial Probability Density Function. IEEE Transactions on Aerospace and Electronic Systems **46**(2), 803–817 (2010) https://doi.org/10.1109/TAES.2010.5461658

[36] Powell, M.J.: A method for nonlinear constraints in minimization problems. Optimization, 283–298 (1969)

[37] Hestenes, M.R.: Multiplier and gradient methods. J. Optim. Theory Appl. **4**(5), 303–320 (1969) https://doi.org/10.1007/BF00927673

[38] Nocedal, J., Wright, S.J.: Numerical Optimization, 2e edn. Springer, New York, NY, USA (2006)

[39] Condat, L.: Fast projection onto the simplex and the l1 ball. Mathematical Programming **158**(1), 575–585 (2016) https://doi.org/10.1007/s10107-015-0946-6

[40] Message Passing Interface Forum: MPI: A Message-Passing Interface Standard Version 4.1. (2023). https://www.mpi-forum.org/docs/mpi-4.1/mpi41-report.pdf

[41] Dagum, L., Menon, R.: OpenMP: An industry-standard API for shared-memory programming. IEEE Comput. Sci. Eng. **5**(1), 46–55 (1998) https://doi.org/10.1109/99.660313

[42] Zhang, L., Datta, A., Coldenstrodt-Ronge, H.B., Jin, X.-M., Eisert, J., Plenio, M.B., Walmsley, I.A.: Recursive quantum detector tomography. New Journal of Physics **14**(11), 115005 (2012) https://doi.org/10.1088/1367-2630/14/11/115005

[43] Chen, X., Xu, F., Xu, H., Zhang, L.: Efficient tomography of coherent optical detectors. Physical Review A **106**(5), 051702 (2022) https://doi.org/10.1103/PhysRevA.106.L051702

[44] Higham, N.J.: Computing a nearest symmetric positive semidefinite matrix. Linear Algebra and its Applications **103**, 103–118 (1988) https://doi.org/10.1016/0024-3795(88)90223-6

[45] Francisco, J.B., Gonçalves, D.S.: A fixed-point method for approximate projection onto the positive semidefinite cone. Linear Algebra and its Applications **523**, 59–78 (2017) https://doi.org/10.1016/j.laa.2017.02.014

[46] Krämer, S., Plankensteiner, D., Ostermann, L., Ritsch, H.: QuantumOptics.jl: A Julia framework for simulating open quantum systems. Computer Physics Communications **227**, 109–116 (2018) https://doi.org/10.1016/j.cpc.2018.02.004

[47] Fousse, L., Hanrot, G., Lefèvre, V., Pélissier, P., Zimmermann, P.: Mpfr: A multiple-precision binary floating-point library with correct rounding. ACM Trans. Math. Softw. **33**(2), 13 (2007) https://doi.org/10.1145/1236463.1236468