npj | Quantum Information

www.nature.com/npjqi

ARTICLE OPEN

Provable compressed sensing quantum state tomography via non-convex methods

Anastasios Kyrillidis[1,2], Amir Kalev[3], Dohyung Park[4], Srinadh Bhojanapalli[5], Constantine Caramanis[6] and Sujay Sanghavi[6]

With nowadays steadily growing quantum processors, it is required to develop new quantum tomography tools that are tailored for high-dimensional systems. In this work, we describe such a computational tool, based on recent ideas from non-convex optimization. The algorithm excels in the compressed sensing setting, where only a few data points are measured from a low-rank or highly-pure quantum state of a high-dimensional system. We show that the algorithm can practically be used in quantum tomography problems that are beyond the reach of convex solvers, and, moreover, is faster and more accurate than other state-of-the-art non-convex approaches. Crucially, we prove that, despite being a non-convex program, under mild conditions, the algorithm is guaranteed to converge to the global minimum of the quantum state tomography problem; thus, it constitutes a provable quantum state tomography protocol.

npj Quantum Information (2018)4:36 ; doi:10.1038/s41534-018-0080-4

INTRODUCTION

Like any other processor, the behavior of a quantum information processor must be characterized, verified, and certified. Quantum state tomography (QST) is one of the main tools for that purpose.[1] Yet, it is generally an inefficient procedure, since the number of parameters that specify quantum states grows exponentially with the number of sub-systems. This inefficiency has two practical manifestations: (i) without any prior information, a vast number of data points needs to be collected;[1] (ii) once the data is gathered, a numerical procedure should be executed on an exponentially high-dimensional space, in order to infer the quantum state that is most consistent with the observations. Thus, to perform QST on steadily growing quantum processors,[2,3] we must introduce novel and efficient techniques for its completion.

Recent advances[4–6] simplify QST by including the premise that, often, our aim is to coherently manipulate *pure* quantum states (i.e., states that can be equivalently described with rank-1, positive semi-definite (PSD) density matrices). The use of such prior information is the *modus operandi* toward making QST more manageable, with respect to the amount of data required.

Compressed sensing (CS)[7] – and its extension to low-rank approximation[8]– has been applied to QST[6,9,10] within this context. Particularly, Gross et al.[6] prove that convex programming guarantees robust estimation of pure n-qubit states from much less information than common approaches require, with overwhelming probability.

These advances, however, leave open the question of how efficiently one can estimate exponentially large-sized quantum states, from a limited set of observations. Since convex programming is susceptible of *provable performance*, typical QST protocols rely on convex programs.[4,6,9] Nevertheless, their weakness remains the high computational and storage complexity. In particular, due to the PSD nature of density matrices, a key step in convex programs is the repetitive application of Hermitian eigensolvers. Such solvers include the well-established family of Lanczos methods,[11–13] the Jacobi-Davinson SVD type of methods,[14] as well as preconditioned hybrid schemes,[15] among others. Since – at least once per iteration – a full eigenvalue decomposition is required in most convex programs, eigensolvers contribute a $\mathcal{O}((2^n)^3)$ computational complexity, where n is the number of qubits of the quantum system. It is obvious that the recurrent application of such eigensolvers makes convex programs impractical, even for quantum systems with a relatively small number n of qubits.[6,16]

Ergo, to improve the efficiency of QST, and of CS QST in particular, we need to complement it with numerical algorithms that can handle large search spaces using limited amount of data, while having rigorous performance guarantees. This is the purpose of this work. Inspired by the recent advances on finding the global minimum in non-convex problems,[17–24] we propose the application of alternating gradient descent for CS QST, that operates directly on the assumed low-rank structure of the density matrix. The algorithm – named Projected Factored Gradient Decent (ProjFGD) – shows significant improvements in QST problems (both in accuracy and efficiency), as compared with state-of-the-art approaches; our numerical experiments justify such behavior.

More crucially, we prove that, despite being a non-convex program, under mild conditions, the algorithm is guaranteed to converge to the global minimum of the QST problem. In general, finding the global minimum in non-convex problems is a hard problem. However, our approach assumes certain regularity conditions – that are, however, satisfied by common CS-inspired protocols in practice[4,6,9] – and a good initialization – which we

[1]IBM T. J. Watson Research Center, 1101 Kitchawan Rd, Yorktown Heights, NY 10598, USA; [2]Department of Computer Science, Rice University, 6100 Main St., Houston, TX 77005-1892, USA; [3]Joint Center for Quantum Information and Computer Science, University of Maryland, College Park, MD 20742, USA; [4]Facebook, 1101 Dexter Ave N, Seattle, WA 98109, USA; [5]Toyota Technological Institute at Chicago, 6045 S Kenwood Ave, Chicago, IL 60637, USA and [6]Department of ECE, University of Texas at Austin, 2501 Speedway, Austin, TX 78712, USA
Correspondence: Anastasios Kyrillidis (anastasios@rice.edu) or Sujay Sanghavi (sanghavi@mail.utexas.edu)

Received: 2 December 2017 Revised: 14 May 2018 Accepted: 22 May 2018
Published online: 01 August 2018

make explicit in the text; both lead to a fast and *provable* estimation of the state of the system, even with limited amount of data.

RESULTS

QST setup

We begin by describing the problem of QST. We are focusing here on QST of a low-rank n-qubit state, ρ_*, from measuring expectation values of n-qubit Pauli observables $\{P_i\}_{i=1}^m$. We denote by $y \in \mathbb{R}^m$ the measurement vector with elements $y_i = \frac{2^n}{\sqrt{m}}\mathrm{Tr}(P_i \cdot \rho_\star) + e_i$, $i = 1, \dots, m$, for some measurement error e_i. The normalization $\frac{2^n}{\sqrt{m}}$ is chosen to follow the results of Liu.[25] For brevity, we denote $\mathcal{M} : \mathbb{C}^{2^n \times 2^n} \to \mathbb{R}^m$ as the linear "sensing" map, such that $(\mathcal{M}(\rho))_i = \frac{2^n}{\sqrt{m}}\mathrm{Tr}(P_i \cdot \rho)$, for $i = 1, \dots, m$.

An n-qubit Pauli observable is given by $P = \otimes_{j=1}^n s_j$ where $s_j \in \{1, \sigma_x, \sigma_y, \sigma_z\}$. There are 4^n such observables in total. In general, one needs to have the expectation values of all 4^n Pauli observables to uniquely reconstruct ρ_*. However, since according to our assumption ρ_* is a low-rank quantum state, we can apply the CS result,[6,25] that guarantees a robust estimation, with high probability, from the measurement of the expectation values of just $m = \mathcal{O}(r2^n n^6)$ randomly chosen Pauli observables, where $r \ll 2^n$ is the rank of ρ_*.

Key property to achieve this is the *restricted isometry property*:[25]

Definition 1 (Restricted Isometry Property (RIP) for Pauli measurements). Let $\mathcal{M} : \mathbb{C}^{2^n \times 2^n} \to \mathbb{R}^m$ be a linear map, such that $(\mathcal{M}(\rho))_i = \frac{2^n}{\sqrt{m}}\mathrm{Tr}(P_i \cdot \rho)$, for $i = 1, \dots, m$. Then, with high probability over the choice of $m = \frac{c}{\delta_r^2} \cdot (r2^n n^6)$ Pauli observables $_{P_i}$, where $c > 0$ is an absolute constant, $\mathcal{M}$ satisfies the r-RIP with constant δ_r, $0 \le \delta_r < 1$; i.e.,

$$(1 - \delta_r)\|\rho\|_F^2 \le \|\mathcal{M}(\rho)\|_2^2 \le (1 + \delta_r)\|\rho\|_F^2,$$

where $\|\cdot\|_F$ *denote the Frobenius norm, is satisfied* $\forall \rho \in \mathbb{C}^{2^n \times 2^n}$ such that $rank(\rho) \le r$.

An accurate estimation of ρ_* is obtained by solving, essentially, a convex optimization problem constrained to the set of quantum states,[9] consistent with the measured data. Among the various problem formulations for QST, two convex program examples are the trace-minimization program that is typically studied in the context of CS QST:

$$\begin{aligned} \underset{\rho \in \mathbb{C}^{2^n \times 2^n}}{\text{minimize}} \quad & \mathrm{Tr}(\rho) \\ \text{subject to} \quad & \rho \succcurlyeq 0, \\ & \|y - \mathcal{M}(\rho)\|_2 \le \varepsilon, \end{aligned} \tag{1}$$

and the least-squares program,

$$\begin{aligned} \underset{\rho \in \mathbb{C}^{2^n \times 2^n}}{\text{minimize}} \quad & \tfrac{1}{2} \cdot \|y - \mathcal{M}(\rho)\|_2^2 \\ & \rho \succcurlyeq 0, \\ \text{subject to} \quad & \mathrm{Tr}(\rho) \le 1, \end{aligned} \tag{2}$$

which is closely related to the (negative) log-likelihood minimization under Gaussian noise assumption. The constraint $\rho \succcurlyeq 0$ captures the positive semi-definite assumption, $\|\cdot\|_2$ is the vector Euclidean ℓ_2-norm, and $\varepsilon > 0$ is a parameter related to the error level in the model. Key in both programs is the combination of the PSD constraint and the trace object: combined, they constitute the tightest convex relaxation to the low-rank, PSD structure of the unknown ρ_*; see also Recht et al.[26]. The constraint $\mathrm{Tr}(\rho) = 1$ is relaxed in Eq. (2) to allow more robustness to noise, following Kalev et al.[9]. The solutions of these programs should be normalized to have unit trace to represent quantum states. We note that if $\mathcal{M}$ corresponds to a positive-operator valued measure (POVM), or includes the identity operator, then the explicit trace constraint is redundant.

As was discussed in the introduction, the problem with convex programs, such as Eqs. (1) and (2), is their inefficiency when applied in high-dimensional systems: most practical solvers for Eqs. (1) and (2) are iterative and handling PSD constraints adds an immense complexity overhead per iteration, especially when n is large.

In this work, we propose to use non-convex programming for QST of low-rank density matrices; we show in practice that it leads to higher efficiency than typical convex programs. We achieve this by restricting the optimization over the intrinsic non-convex structure of rank-r PSD matrices. This allow us to "describe" an $2^n \times 2^n$ PSD matrix with only $\mathcal{O}(2^n r)$ space, as opposed to the $\mathcal{O}((2^n)^2)$ ambient space. Even more substantially, our program has theoretical guarantees of global convergence, similar to the guarantees of convex programming, while maintaining faster performance than the latter. These properties make our scheme ideal to complement the CS methodology for QST in practice.

Projected factored gradient descent algorithm

Optimization criterion recast. At its basis, the Projected Factored Gradient Descent (ProjFGD) algorithm transforms convex programs, such as in Eqs. (1)–(2), by enforcing the factorization of a $d \times d$ PSD matrix ρ such that $\rho = AA^\dagger$, where $d = 2^n$. This factorization, popularized by Burer and Monteiro[27] for solving semi-definite convex programming instances, naturally encodes the PSD constraint, removing the expensive eigen-decomposition projection step. For concreteness, we focus here on the convex program (Eq. (2)). In order to encode the trace constraint, ProjFGD enforces additional constraints on A. In particular, the requirement that $\mathrm{Tr}(\rho) \le 1$ is equivalently translated to the *convex* constraint $\|A\|_F^2 \le 1$, where $\|\cdot\|_F$ is the Frobenius norm. The above recast the program (Eq. (2)) as a non-convex program:

$$\begin{aligned} \underset{A \in \mathbb{C}^{d \times r}}{\text{minimize}} \quad & f(AA^\dagger) := \tfrac{1}{2} \cdot \left\| y - \mathcal{M}(AA^\dagger) \right\|_2^2 \\ \text{subject to} \quad & \|A\|_F^2 \le 1. \end{aligned} \tag{3}$$

Given $\mathrm{rank}(\rho_*) = r$, programs Eqs. (2) and (3) are equivalent in the sense that the optimal value of Eq. (2) is identical to that of Eq. (3), by the relation $\rho = AA^\dagger$; however, program Eq. (3) might have additional local solutions. Further, while the constraint set is convex, the objective is no longer convex due to the bilinear transformation of the parameter space $\rho = AA^\dagger$. Such criteria have been studied recently in machine learning and signal processing applications.[17–24] Here, the added twist is the inclusion of further matrix norm constraints, that makes it proper for tasks such as QST; as we show in the Supplementary information Section A, such addition complicates the algorithmic analysis.

The ProjFGD algorithm and its guarantees. At heart, ProjFGD is a projected gradient descent algorithm over the variable A; i.e.,

$$A_{t+1} = \Pi_{\mathcal{C}}\left(A_t - \eta \nabla f\left(A_t A_t^\dagger\right) \cdot A_t\right),$$

where $\Pi_{\mathcal{C}}(B)$ denotes the projection of a matrix $B \in \mathbb{C}^{d \times r}$ onto the set $\mathcal{C} = \{A : A \in \mathbb{C}^{d \times r}, \|A\|_F^2 \le 1\}$. $\nabla f(\cdot) : \mathbb{R}^{d \times d} \to \mathbb{R}^{d \times d}$ denotes the gradient of the function f. Specific details of the ProjFGD algorithm, along with a pseudocode implementation, are provided in the Method Section and in the Supplementary information Sections A and B. Here, we focus on the theoretical guarantees of the ProjFGD. In summary, our theory dictates a specific *constant* step-size selection, η, that guarantees convergence to the global minimum, assuming a satisfactory initial point ρ_0 is provided.

An important issue in optimizing Eq. (3) over the factored space is the existence of non-unique possible factorizations for a given ρ. To see this, if $\rho = AA^\dagger$, then for any unitary matrix $R \in \mathbb{C}^{r\times r}$ such that $RR^\dagger = I$, we have $\rho = \widehat{A}\widehat{A}^\dagger$, where $\widehat{A} = AR$. Since we are interested in obtaining a low-rank solution in the original space, we need a notion of distance to ρ_* over the factors. We use the following unitary-invariant distance metric: Let matrices

Definition 2. Let matrices $A, A_\star, \in \mathbb{C}^{d\times r}$. Define:

$$\mathrm{DIST}(A, A_\star) := \min_{R:R\in\mathcal{U}} \|A - A_\star R\|_F,$$

where $\mathcal{U}$ is the set of $r\times r$ unitary matrices.

Let us first describe the local convergence rate guarantees of ProjFGD.

Theorem 3 (Local convergence rate for QST). Let $\rho_\star$ be a rank-r quantum state density matrix of an n-qubit system with a non-unique factorization $\rho_\star = A_\star A_\star^\dagger$, *for* $A_\star \in \mathbb{C}^{2^n\times r}$. *Let* $y \in \mathbb{R}^m$ be the measurement vector of $m = \mathcal{O}(rn^6 2^n)$ random n-qubit Pauli observables, and $\mathcal{M}$ be the corresponding sensing map, such that $y_i = (\mathcal{M}(\rho_\star))_i + e_i$, $\forall i = 1, \ldots, m$. Let the step η in ProjFGD satisfy:

$$\eta \le \frac{1}{128\left(\widehat{L}\sigma_1(\rho_0)+\sigma_1(\nabla f(\rho_0))\right)}, \tag{4}$$

where $\sigma_1(\rho)$ denotes the leading singular value of ρ. Here, $\widehat{L} \in (1, 2)$ and $\rho_0 = A_0 A_0^\dagger$ is the initial point such that:

$$\mathrm{DIST}(A_0, A_\star) \le \gamma' \sigma_r(A_\star),$$

for $\gamma' := c \cdot \frac{(1-\delta_{4r})}{(1+\delta_{4r})} \cdot \frac{\sigma_r(\rho_\star)}{\sigma_1(\rho_\star)}$, $c \le \frac{1}{200}$, where δ_{4r} is the RIP constant. Let A_t be the estimate of ProjFGD at the t-th iteration; then, the new estimate A_{t+1} satisfies

$$\mathrm{DIST}(A_{t+1}, A_\star)^2 \le \alpha \cdot \mathrm{DIST}(A_t, A_\star)^2, \tag{5}$$

where $\alpha := 1 - \frac{(1-\delta_{4r})\cdot\sigma_r(\rho_\star)}{550\left((1+\delta_{4r})\sigma_1(\rho_\star)+\|e\|_2\right)} < 1$. Further, A_{t+1} satisfies $\mathrm{DIST}(A_{t+1}, A_\star) \le \gamma'\sigma_r(A_\star)$, $\forall t$.

The proof of Theorem 3 is provided in the Supplementary information Section A. The definitions of L and $\widehat{L}$ can be found in the Methods Section; for our discussion, they can be assumed constants. The above theorem provides a *local* convergence guarantee: given an initialization point $\rho_0 = A_0 A_0^\dagger$ close enough to the optimal solution –in particular, where $\mathrm{DIST}(A_0, A_\star) \le \gamma'\sigma_r(A_\star)$ is satisfied– our algorithm converges locally with linear rate. In order to obtain $(A_T, A_\star)^2 \le \varepsilon$, ProjFGD requires $T = \mathcal{O}\left(\log\frac{\gamma'\cdot\sigma_r(A_\star)}{\varepsilon}\right)$ number of iterations. We conjecture that this further translates into linear convergence in the infidelity metric, $1 - \left(\sqrt{\sqrt{\rho_T}\rho \star \sqrt{\rho_T}}\right)^2$.

So far, we assumed ρ_0 is provided such that $\mathrm{DIST}(A_0, A_\star) \le \gamma'\sigma_r(A_\star)$. The next theorem proposes an initialization procedure that could achieve this guarantee (under assumptions) and turns the above local guarantees to convergence to the global minimum.

Lemma 4. Let A_0 be such that $\rho_0 = A_0 A_0^\dagger = \Pi_{\mathcal{C}'}\left(\frac{-1}{L}\cdot\nabla f(0)\right)$, where $\Pi_{\mathcal{C}'}(\cdot)$ is the projection onto the set of PSD matrices ρ that satisfy $Tr(\rho) \le 1$, and $\nabla f(0)$ denotes the gradient of f evaluated at the all zero matrix. Consider the problem (3) where $\mathcal{M}$ satisfies the RIP for some constant $\delta_{4r} \in (0, 1)$. Further, assume the optimum point $\rho_\star$ satisfies $rank(\rho_\star) = r$. Then, A_0 satisfies:

$$\mathrm{DIST}(A_0, A_\star) \le \gamma' \cdot \sigma_r(A_\star),$$

where $\gamma' = \sqrt{\frac{1-\frac{1-\delta_{4r}}{1+\delta_{4r}}}{2(\sqrt{2}-1)}} \cdot \tau(\rho_\star) \cdot \sqrt{srank(\rho_\star)}$ and $srank(\rho) = \frac{\|\rho\|_F}{\sigma_1(\rho)}$.

The proof is provided in Supplementary information Section B. This initialization introduces further restrictions on the condition number of $\rho_\star$, $\tau(\rho_\star) = \frac{\sigma_1(\rho_\star)}{\sigma_r(\rho_\star)}$, and the condition number of the objective function, which is proportional to $\propto \frac{1+\delta_{4r}}{1-\delta_{4r}}$. The initialization assumptions in Theorem 3 are satisfied by Lemma 4 if $\mathcal{M}$ satisfies RIP with a constant δ_{4r} fulfilling the following condition:

$$\frac{1+\delta_{4r}}{1-\delta_{4r}} \cdot \sqrt{1 - \frac{1-\delta_{4r}}{1+\delta_{4r}}} \le \frac{\sqrt{2\left(\sqrt{2}-1\right)}}{200} \cdot \frac{1}{\sqrt{r}\cdot\tau^2(\rho_\star)}. \tag{6}$$

In the special case of $r = 1$, $\tau(\rho_\star) = 1$ and, $\mathrm{srank}(\rho_\star) = 1$, the condition simplifies to $\delta_{4r} \lesssim 10^{-5}$. While these conditions are hard to check a priori, Pauli observables satisfy them, with high probability, as n increases, according to the results of Liu.[25]

In summary, we have shown that, with a proper initialization and a constant step size, the ProjFGD algorithm converges to the global minimum, if the sensing map satisfies the RIP with a small constant, according to Eq. (6). This condition is satisfied, with high probability, by a measurement of $\mathcal{O}(rn^6 2^n)$ random Pauli observables.

We note that the conditions for global convergence are sufficient but not necessary. As we shall see in the experiments below, we obtain convergence to the global minimum (or to a point very close to it) with milder conditions, such as random initialization. Moreover, recent advances in machine learning[22] have shown that, under RIP, random initialization guarantees global convergence of a variant of our algorithm, where we exclude the trace constraint in Eq. (2). This is the case where $\mathcal{M}$ corresponds to a POVM, or includes the identity operator.

Numerical experiments evaluation

Our experiments follow the discussion above. We find that our initialization, as well as random initialization, works well in practice, and this behavior has been observed repeatedly in all the experiments we conducted. Thus, the method returns the exact solution of the convex programming problem, while being orders of magnitude faster than state-of-the-art optimization programs.

In all the experiments, the error is reported in the Frobenius metric, $\|\widehat{\rho} - \rho_\star\|_F / \|\rho_\star\|_F$, where $\widehat{\rho}$ is the estimation of the true state $\rho_\star$. Note that for a pure state ρ, $\|\rho\|_F = 1$. For some experiments we also report the infidelity metric $1 - \mathrm{Tr}\left(\sqrt{\sqrt{\rho_\star}\widehat{\rho}\sqrt{\rho_\star}}\right)^2$. We model the additive noise in our experiments, $e \in \mathbb{R}^m$, according to a circularly-symmetric normal distribution with variance σ for each measurement, $e \sim \mathcal{CN}(0, \sigma \cdot I)$.

Comparison of ProjFGD with second-order methods. As a first set of experiments, we compare the efficiency of ProjFGD with *second-order* cone convex programs. State-of-the-art solvers within this class of solvers are the SeDuMi and SDPT3 methods; for their use, we rely on the off-the-shelf Matlab wrapper CVX.[28] In our experiments, we observed that SDPT3 was faster and we select it for our comparison. The setting is as described in the Results Section, where additive noise has variance σ, i.e., $\sim \mathcal{CN}(0, \sigma \cdot I)$. We consider both convex formulations Eqs. (1)–(2) and compare it to the ProjFGD estimator with $r = 1$; in figures we use the notation CVX 1 and CVX 2 for simplicity.

We consider two cases: (i) $n = 7$, and (ii) $n = 13$. Table 1 shows median values of ten independent experimental realizations for $m = \frac{7}{3} rd \log d$; this selection of m was made so that all algorithms return a solution close to the optimum $\rho_\star$. Empirically, we have observed that ProjFgD succeeds even for cases $m = \mathcal{O}(rd)$. We consider both noiseless $\sigma = 0$ and noisy $\sigma = 0.05$ settings.

Table 1. All values are median values over ten independent Monte Carlo iterations. "N/A" indicates that the corresponding algorithms did not return a solution within the selected wall-time T. We set $T = 86400$ s (1 day)

Algorithm	$d = 2^7$					$d = 2^{13}$				
	$\sigma = 0$		$\sigma = 0.05$			$\sigma = 0$		$\sigma = 0.05$		
	Time [s]	$\frac{\|\hat{\rho}-\rho_\star\|_F}{\|\rho_\star\|_F}$	Time [s]	$\frac{\|\hat{\rho}-\rho_\star\|_F}{\|\rho_\star\|_F}$	Infidelity	Time [s]	$\frac{\|\hat{\rho}-\rho_\star\|_F}{\|\rho_\star\|_F}$	Time [s]	$\frac{\|\hat{\rho}-\rho_\star\|_F}{\|\rho_\star\|_F}$	Infidelity
(1)	46.01	5.3538e-07	58.48	6.0405e-02	3.0394e-02	N/A	N/A	N/A	N/A	N/A
(2)	77.12	3.0645e-04	65.53	6.1407e-02	3.0559e-02	N/A	N/A	N/A	N/A	N/A
ProjFGD	0.28	3.2224e-08	0.30	2.3540e-02	1.3820e-04	1314.01	6.8469e-08	1487.22	3.1104e-02	1.9831e-03

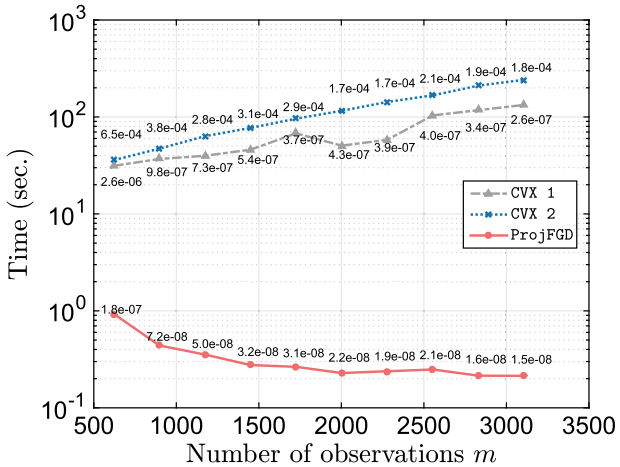

Fig. 1 Dimension fixed to $d = 2^7$ with $\text{rank}(\rho_*) = 1$. The figure depicts the noiseless setting. Numbers within figure are the error in Frobenius norm achieved (median values)

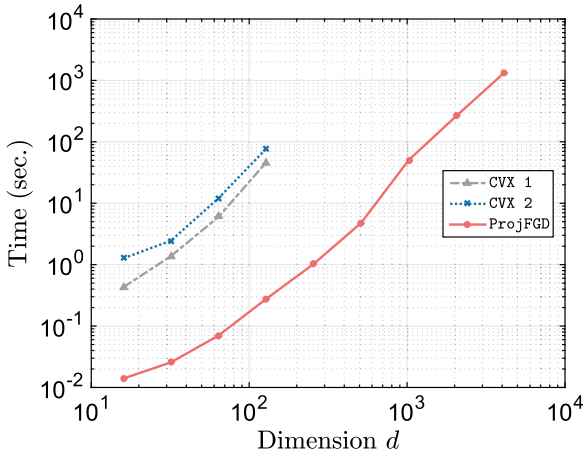

Fig. 2 Number of data points set to $m = \frac{7}{3} rd \log d$. Rank of optimum point is set to $\text{rank}(\rho_*) = 1$. The figure depicts the noiseless setting

Figures 1 and 2 show graphically how second-order convex vs. our first-order non-convex schemes scale. In Fig. 1, we observe that, while in the ProjFGD more observations lead to faster convergence,[29] the same does not hold for the second-order cone programs. In Fig. 2, it is obvious that the convex solvers do not scale easily beyond $n = 7$, whereas our method handles cases up to $n = 13$, within reasonable time. We note that, as n increases, a significant amount of time in our algorithm is spent forming the Pauli measurement vectors $_{Pi}$; i.e., assuming that the application of

Table 2. Median results for reconstruction and efficiency, for $n = 13$ qubits and $C_{\text{sam}} = 3$

Algorithm	$\frac{\|\hat{\rho}-\rho_\star\|_F}{\|\rho_\star\|_F}$	Time [s]
AccUniPDGrad	7.4151e-02	2354.4552
ProjFGD	8.6309e-03	1214.0654

$_{Pi}$'s takes the same amount of time as in CVX solvers, ProjFGD requires much less additional computational power per iteration, compared with CVX 1 and CVX 2.

Comparison of ProjFGD with first-order methods. We compare our method with more efficient first-order methods, both convex (AccUniPDGrad[30]) and non-convex (SparseApproxSDP[31] and RSVP[32]); we briefly describe these methods in the Discussion Section.

We consider two settings: $\rho_\star$ is (i) a pure state (i.e., $\text{rank}(\rho_\star) = 1$) and, (ii) a nearly low-rank state. In the latter case, we construct $\rho_\star = \rho_{\star,r} + \zeta$, where $\rho_{\star,r}$ is a rank-deficient PSD satisfying $\text{rank}(\rho_{\star,r}) = r$, and $\zeta \in \mathbb{C}^{d\times d}$ is a full-rank PSD noise term with a fast decaying eigen-spectrum, significantly smaller than the leading eigen values of $\rho_{\star,r}$. In other words, we can well-approximate $\rho_\star$ with $\rho_{\star,r}$. For all cases, the noise is such that $\|e\| = 10^{-3}$. The number of data points m satisfy $m = C_{\text{sam}} \cdot rd$, for various values of C_{sam}>0.

Table 2 contains recovery error and execution time results for the case $n = 13$ ($d = 8192$); in this case, we solve a $d^2 = 67,108,864$ dimensional problem. For this case, RSVP and SparseApproxSDP algorithms were excluded from the comparison, due to excessive execution time. Supplementary information Section C provides extensive results, where similar performance is observed for other values of $d = 2^n$ and C_{sam}.

Table 3 considers the more general case where $\rho_\star$ is nearly low-rank: i.e., it can be well-approximated by a density matrix $\rho_{\star,r}$ where $r = 20$ (low-rank density matrix). In this case, $n = 12$ ($d = 4096$), $m = 245{,}760$ for $C_{\text{sam}} = 3$. As the rank in the model, r, increases, algorithms that utilize an SVD routine spend more CPU time on singular value/vector calculations. Certainly, the same applies for matrix-matrix multiplications; however, in the latter case, the complexity scale is milder than that of the SVD calculations. For completeness, in Supplementary information Section C we provide results that illustrate the effect of random initialization: Similar to above, ProjFGD shows competitive behavior by finding a better solution faster, irrespective of initialization point.

Overall, ProjFGD shows a substantial improvement in performance, as compared to the state-of-the-art algorithms; we would like to emphasize that projected gradient descent schemes, such as in Becker et al.,[32] are also efficient in small- to medium-sized problems, due to their fast convergence rate. Further, convex approaches might show better sampling complexity performance (i.e., as C_{sam} decreases). Nevertheless, one can perform accurate

Table 3. Median results for reconstruction and efficiency

Algorithm	Setting: $r = 5$.		Setting: $r = 20$.	
	$\frac{\|\rho - \rho_{\star,r}\|_F}{\|\rho_{\star,r}\|_F}$	Time [s]	$\frac{\|\rho - \rho_{\star,r}\|_F}{\|\rho_{\star,r}\|_F}$	Time [s]
SparseApproxSDP	1.65e-02	7029.15	4.08e-02	7514.13
RSVP	2.36e-01	6314.42	3.69e-02	7341.92
AccUniPDGrad	4.40e-02	231.10	5.23e-02	452.05
ProjFGD	4.23e-02	126.41	4.73e-03	220.71

Time reported is in seconds. $C_{\text{sam}} = 3$ and $n = 12$.

maximum-likelihood estimation for larger systems in the same amount of time using our methods for such small- to medium-sized problems. We defer the reader to Supplementary information Section C, due to space restrictions.

DISCUSSION

In this work, we propose a non-convex algorithm, dubbed as ProjFGD, for estimating a highly-pure quantum state, in a high-dimensional Hilbert space, from relatively small number of data points. We showed empirically that ProjFGD is orders of magnitude faster than state-of-the-art convex and non-convex programs, such as Yurtsever et al.,[30] Hazan,[31] and Becker et al.[32]. More importantly, we prove that under proper initialization and step size, the ProjFGD is guaranteed to converge to the global minimum of the problem, thus ensuring a provable tomography procedure; see Theorem 3 and Lemma 4.

Our techniques and proofs can be applied to scenaria beyond the ones considered in this work. We conjecture that our results apply for other "sensing" settings, that are informationally complete for low-rank states; see e.g., Baldwin et al.[4]. The results presented here are independent of the noise model and could be applied for non-Gaussian noise models, such as those stemming from finite counting statistics. Lastly, while here we focus on state tomography, it would be interesting to explore similar techniques for the problem of process tomography.

Related work

In order to place our work in the literature, we focus on several efficient methods for QST; for a broader set of citations that go beyond QST, see Park et al.[21]

The use of non-convex algorithms in QST is not new, and dates before the introduction of the CS protocol in QST settings.[6] Even the use of the reparameterization $\rho = AA^\dagger$ is not new; see the works.[33–36] Albeit their success, there are no theoretical results on the non-convex nature of the transformed objective (e.g., the presence of spurious local minima), except for the case of Goncalves et al..[37] In that work, the authors consider the *informationally complete* case, where the number of measurements is of the order $\mathcal{O}(d^2)$, and therefore, there is a unique solution in Eqs. (1)–(2), without the requirement of the RIP. The authors characterize the local vs. the global behavior of the objective under the factorization $\rho = AA^\dagger$ and discuss how existing methods fail due to improper stopping criteria or due to the lack of algorithmic convergence results. Their work highlights the lack of rigorous convergence results of algorithms used in QST.

Shang et al.[38] propose a hybrid algorithm that (i) starts with a conjugate-gradient (CG) algorithm in the A space, in order to get initial rapid descent, and (ii) switch over to accelerated first-order methods in the original ρ space, provided one can determine the switchover point cheaply. Under the multinomial maximum-likelihood objective, in the initial CG phase, the Hessian of the objective is computed per iteration (i.e., a $d^2 \times d^2$ matrix), along with its eigenvalue decomposition. Such an operation is costly, even for moderate values of d, and heuristics are proposed for its completion. From a theoretical perspective, Shang et al.[38] provide no convergence or convergence rate guarantees.

Goncalves et al.[39] the authors study the QST problem in the original parameter space, and propose a projected gradient descent algorithm. The proposed algorithm applies both in convex and non-convex objectives, and convergence only to stationary points could be expected. Bolduc et al.[40] extends the work of Goncalves et al.[39] with two first-order variants, using momentum motions, similar to the techniques proposed by Polyak and Nesterov for faster convergence in convex optimization.[41] The above algorithms operate in the informationally complete case. Similar ideas in the informationally incomplete case can be found in these works.[32,42]

Very recently, Riofrio et al.[42] presented an experimental implementation of CS tomography of a $n = 7$ qubit system, where only 127 Pauli basis measurements are available. To achieve recovery in practice, the authors proposed a computationally efficient estimator, based on the factorization $\rho = AA^\dagger$. The resulting method resembles our gradient descent method on the factors A. d: However, the authors focus only on the experimental efficiency of the method and provide no specific results on the optimization efficiency of the algorithm, what are its theoretical guarantees, and how its components (such as initialization and step size) affect its performance (e.g., the step size is set to a sufficiently small constant). See also Schwemmer et al.[10] for a six-qubit implementation.

One of the first provable algorithmic solutions for the QST problem was through convex approximations:[26] this includes nuclear norm minimization approaches,[6] as well as proximal variants, as the one that follows:

$$\underset{\rho \succcurlyeq 0}{\text{minimize}} \quad \|y - \mathcal{M}(\rho)\|_F^2 + \lambda \text{Tr}(\rho). \tag{7}$$

See Gross et al.[6] for the theoretical analysis. Within this context, we mention the work of Yurtsever et al.:[30] there, the AccUniPDGrad algorithm is proposed – a universal primal-dual convex framework with sharp operators, in lieu of proximal low-rank operators – where QST is considered as an application. AccUniPDGrad combines the flexibility of proximal primal-dual methods with the computational advantages of conditional gradient methods.

Hazan[31] presents SparseApproxSDP algorithm that solves the QST problem in Eq. (2), when the objective is a generic gradient Lipschitz smooth function, by updating a putative low-rank solution with rank-1 refinements, coming from the gradient. This way, SparseApproxSDP avoids computationally expensive operations per iteration, such as full eigen-decompositions. In theory, SparseApproxSDP achieves a sublinear $O\left(\frac{1}{\varepsilon}\right)$ convergence rate. However, depending on ε, SparseApproxSDP might not return a low-rank solution.

Finally, Becker et al.[32] propose Randomized Singular Value Projection (RSVP), a projected gradient descent algorithm for QST, which merges gradient calculations with truncated eigen-decompositions, via randomized approximations for computational efficiency.

Future directions. We conclude with a short list of interesting future research directions. Our immediate goal is the application of ProjFGD in real-world scenaria; this could be completed by utilizing IBM quantum computers.[3] This complements the results found in Riofrio et al.[43] for a different quantum system.

Beyond its use as point estimator, the maximum-likelihood estimator is used as a basis for inference around the point estimate, via confidence intervals[44] and credible regions.[45] However, there is still no rigorous analysis when the factorization $\rho = AA^\dagger$ is used.

The work in refs. [38,40] considers accelerated gradient descent methods for QST in the original parameter space ρ. It remains an open question how our approach could exploit such techniques, along with rigorous approximation and convergence guarantees. Further, distributed/parallel implementations, like Hou et al.,[46] remain widely open using our approach, in order to accelerate further the execution of the algorithm. Research along these directions is very interesting and is left for future work.

Finally, we identify two practical observations from our experiments that need further theoretical justification. First, we saw numerically that a random initialization in our settings works well; a careful theoretical treatment for this case is an open problem. Second, while we observed that the ProjFGD outperforms convex solvers; it is an open question to understand its behavior in the setting where $r = d$.

METHODS

Next follows a more detailed discussion on ProjFGD. The pseudocode is provided in Algorithm 1; a real implementation is in Supplementary information Section C.

Algorithm 1 ProjFGD pseudocode for (3)

Input: Function f, target rank r, # iterations T.

Output: $\rho = A_T A_T^\dagger$.

Initialize ρ_0 randomly or set $\rho_0 := {}^2/_{\hat{L}} \cdot \Pi_{\mathcal{C}'}(\mathcal{M}^*(y))$.

Set $A_0 \in \mathbb{C}^{d\times r}$ such that $\rho_0 = A_0 A_0^\dagger$.

Set step size η as in (4).

for $t = 0$ to $T-1$ **do**

$A_{t+1} = \Pi_{\mathcal{C}}\left(A_t - \eta \nabla f(A_t A_t^\dagger)\cdot A_t\right)$.

end

Denote $g(A) = \frac{1}{2}\cdot\|y - \mathcal{M}(AA^\dagger)\|_2^2$ and $f(\rho) = \frac{1}{2}\cdot\|y - \mathcal{M}(\rho)\|_2^2$. Due to the symmetry of f, i.e., $f(\rho) = f(\rho^\dagger)$, the gradient of $g(A)$ w.r.t. A variable is given by

$$\nabla g(A) = \left((\rho) + (\rho)^\dagger\right)\cdot A = 2(\rho)\cdot A,$$

where $\nabla f(\rho) = -2\mathcal{M}^*(y - \mathcal{M}(\rho))$, and $\mathcal{M}^*$ is the adjoint operator for $\mathcal{M}$. For the Pauli measurements case we consider in this paper, the adjoint operator for an input vector $b \in \mathbb{R}^m$ is $\mathcal{M}^*(b) = \frac{2^n}{\sqrt{m}}\sum_{i=1}^m b_i P_i$.

The prior knowledge $\text{rank}(\rho_\star) \le r_\star$ is imposed by setting $A \in \mathbb{C}^{d\times r_\star}$. In real experiments, the state $\rho_\star$ could be full rank, but often is highly-pure with only few dominant eigenvalues.[43] In this case, $\rho_\star$ is well-approximated by a low-rank matrix of rank r, which can be much smaller than $r_\star$. In the ProjFGD protocol, we set $A \in \mathbb{C}^{d\times r}$. In this form, A contains far fewer variables to maintain and optimize than a $d\times d$ PSD matrix, and thus it is easier to update and to store its iterates.

The per-iteration complexity of ProjFGD is dominated by the application of the linear map $\mathcal{M}$ and by matrix–matrix multiplications. While both eigenvalue decomposition and matrix multiplication have $\mathcal{O}\left((2^n)^2 r\right)$ complexity, the latter is at least two-orders of magnitude faster on dense matrices.[21]

Due to the bilinear structure in Eq. (3), it is not clear whether the factorization $\rho = AA^\dagger$ introduces *spurious* local minima, i.e., minima that do not exist in Eqs. (1)–(2), but are "created" after the factorization. This necessitates careful initialization to obtain the global minimum.

The initial point ρ_0 is set as $\rho_0 := 1/\hat{L}\cdot\Pi_{\mathcal{C}'}(-\nabla f(0)) = 2/\hat{L}\cdot\Pi_{\mathcal{C}'}(\mathcal{M}^*(y))$, where $\Pi_{\mathcal{C}'}(\cdot)$ denotes the projection onto the set of PSD matrices ρ that satisfy $\text{Tr}(\rho)\le 1$. Here, $\hat{L}$ represents an approximation of L, where L is such that for all rank-r matrices ρ, ζ:

$$\|\nabla f(\rho) - \nabla f(\zeta)\|_F \le L\cdot\|\rho - \zeta\|_F. \tag{8}$$

(This also means that f is *restricted gradient Lipschitz continuous* with parameter L. We defer the reader to the Supplementary information Sections A and B for more information). In practice, we set $\hat{L}\in(1,2)$.

This is the only place where eigenvalue-type calculation is required. The projection $\Pi_{\mathcal{C}'}(\cdot)$ is given in ref. [39]. In practice, we could just use a standard projection onto the set of PSD matrices $\rho_0 := 2/\hat{L}\cdot\Pi_+(\mathcal{M}^*(y))$; our numerical experiments show that it is sufficient and can be implemented by any off-the-shelf eigenvalue solver. In that case, the algorithm generates $A_0 \in \mathbb{C}^{d\times r}$ by truncating the computed eigen-decomposition, followed by a projection onto the convex set, $\mathcal{C}$.

Data availability

The empirical results were obtained via synthetic experiments; the algorithm's implementation is available in the supplementary material.

ACKNOWLEDGEMENTS

Anastasios Kyrillidis is supported by the IBM Goldstine Fellowship and Amir Kalev is supported by the DoD.

AUTHOR CONTRIBUTIONS

All authors have made substantial contributions to the paper: design of the work, drafting the manuscript, final approval and accountability for all aspects of the work. A. Kyrillidis performed the calculations and conducted the numerical experiments.

ADDITIONAL INFORMATION

Supplementary information accompanies the paper on the *npj Quantum Information* website (https://doi.org/10.1038/s41534-018-0080-4).

Competing interests: The authors declare no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Change history: In the original published HTML version of this Article, some of the characters in the equations were not appearing correctly. This has now been corrected in the HTML version.

REFERENCES

1. Altepeter, J., Jeffrey, E. & Kwiat, P. Photonic state tomography, advances in atomic. *Mol. Opt. Phys.* **52**, 105–159 (2005).
2. Zhang, J. et al. Observation of a many-body dynamical phase transition with a 53-qubit quantum simulator. Quantum Phys. Preprint at https://arxiv.org/abs/1708.01044 (2017).
3. IBM-Q: Quantum computing research. https://www.research.ibm.com/ibm-q/ (2018).
4. Baldwin, C. H., Deutsch, I. H. & Kalev, A. Strictly-complete measurements for bounded-rank quantum-state tomography. *Phys. Rev. A* **93**(5), 052105 (2016).
5. Flammia, S., Silberfarb, A. & Caves, C. Minimal informationally complete measurements for pure states. *Quantum Phys.* **35**(12), 1985–2006 (2005).
6. Gross, D., Liu, Y.-K., Flammia, S. T., Becker, S. & Eisert, J. Quantum state tomography via compressed sensing. *Phys. Rev. Lett.* **105**(15), 150401 (2010).
7. Donoho, D. Compressed sensing. *IEEE Trans. Inf. Theory* **52**(4), 1289–1306 (2006).
8. Candès, E. & Recht, B. Exact matrix completion via convex optimization. *Found. Comput. Math.* **9**(6), 717 (2009).
9. Kalev, A., Kosut, R. & Deutsch, I. Quantum tomography protocols with positivity are compressed sensing protocols. *NPJ Quantum Inf.* **1**, 15018 (2015).
10. Schwemmer, C. et al. Experimental comparison of efficient tomography schemes for a six-qubit state. *Phys. Rev. Lett.* **113**(4), 040503 (2014).
11. Baglama, J. & Reichel, L. Restarted block Lanczos bidiagonalization methods. *Numer. Algorithms* **43**(3), 251–272 (2006).
12. Cullum, J., Willoughby, R. & Lake, M. A Lanczos algorithm for computing singular values and vectors of large matrices. *SIAM J. Sci. Stat. Comput.* **4**(2), 197–215 (1983).
13. Kokiopoulou, E., Bekas, C. & Gallopoulos, E. Computing smallest singular triplets with implicitly restarted Lanczos bidiagonalization. *Appl. Numer. Math.* **49**(1), 39–61 (2004).
14. Hochstenbach, M. A Jacobi–Davidson type SVD method. *SIAM J. Sci. Comput.* **23**(2), 606–628 (2001).
15. Wu, L. & Stathopoulos, A. A preconditioned hybrid SVD method for accurately computing singular triplets of large matrices. *SIAM J. Sci. Comput.* **37**(5), S365–S388 (2015).
16. Haffner, H. et al. Scalable multi-particle entanglement of trapped ions. *Nature* **438**, 643–646 (2006).

17. Bhojanapalli, S., Kyrillidis, A. & Sanghavi, S. Dropping convexity for faster semi-definite op- timization. In *29th Annual Conference on Learning Theory, Proceedings of Machine Learning Research*. **49**, 530–582 (2016).
18. Chen, Y. & Wainwright, M. Fast low-rank estimation by projected gradient descent: general statistical and algorithmic guarantees. Preprint at https://arxiv.org/abs/1509.03025 (2015).
19. Ge, R., Lee, J. & Ma, T. Matrix completion has no spurious local minimum, In *Advances in Neural Information Processing Systems*, 2973–2981 (2016).
20. Park, D., Kyrillidis, A., Bhojanapalli, S., Caramanis, C. & Sanghavi, S. Provable Burer–Monteiro factorization for a class of norm-constrained matrix problems. Preprint at https://arxiv.org/abs/1606.01316 (2016).
21. Park, D., Kyrillidis, A., Caramanis, C. & Sanghavi, S. Finding low-rank solutions to matrix problems, eficiently and provably. Preprint at https://arxiv.org/abs/1606.03168 (2016).
22. Park, D., Kyrillidis, A., Carmanis, C. & Sanghavi, S. Non-square matrix sensing without spurious local minima via the Burer–Monteiro approach. In *Artificial Intelligence and Statistics*, 65–74 (2016).
23. Sun, R. & Luo, Z.-Q. Guaranteed matrix completion via nonconvex factorization. In *IEEE Annual Symposium on Foundations of Computer Science*, 270–289 (2015).
24. Tu, S., Boczar, R., Simchowitz, M., Soltanolkotabi, M. & Recht, B. Low-rank solutions of linear matrix equations via Procrustesow. In *Proceedings of International Conference on International Conference on Machine Learning*. **48**, 964–973 (2015).
25. Liu, Y.-K. Universal low-rank matrix recovery from Pauli measurements. In *Advances in Neural Information Processing Systems*, 1638–1646 (2011).
26. Recht, B., Fazel, M. & Parrilo, P. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.* **52**(3), 471–501 (2010).
27. Burer, S. & Monteiro, R. D. C. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Math. Program.* **95**, 329–357 (2003).
28. CVX Research, CVX: Matlab software for disciplined convex programming, version 2.0. http://cvxr.com/cvx (2012).
29. Chandrasekaran, V. & Jordan, M. Computational and statistical tradeoffs via convex relax-ation. *Proc. Natl Acad. Sci.* **110**(13), E1181–E1190 (2013).
30. Yurtsever, A., Dinh, Q. T. & Cevher, V. A universal primal-dual convex optimization frame-work. In *Advances in Neural Information Processing Systems*, 3150–3158 (2015).
31. Hazan, E. Sparse approximate solutions to semidefinite programs. *Lect. Notes Comput. Sci.* **4957**, 306–316 (2008).
32. Becker, S., Cevher, V. & Kyrillidis, A. Randomized low-memory singular value projection. In *10th International Conference on Sampling Theory and Applications* (*Sampta*), 364–367 (2013).
33. Banaszek, K., D'Ariano, G. M., Paris, M. G. A. & Sacchi, M. F. Maximum-likelihood estimation of the density matrix. *Phys. Rev. A* **61**(1), 010304 (1999).
34. Paris, M., D'Ariano, G. & Sacchi, M. Maximum-likelihood method in quantum estimation. *AIP Conf. Proc.* **568**, 456–467 (2001).
35. Teo, Y. S., Rehácek, J. & Hradil, Z. Informationally incomplete quantum tomography. *Quantum Meas. Quantum Metrol.* **1**, 57–83 (2013).
36. Rehácek, J., Hradil, Z., Knill, E. & Lvovsky, A. I. Diluted maximum-likelihood algorithm for quantum tomography. *Phys. Rev. A* **75**, 042108 (2007).
37. Goçalves, D., Gomes-Ruggiero, M., Lavor, C., Farias, O. J. & Ribeiro, P. Local solutions of maximum likelihood estimation in quantum state tomography. *Quantum Inf. & Comput.* **12**(9-10), 775–790 (2012).
38. Shang, J., Zhang, Z. & Ng, H. K. Superfast maximum-likelihood reconstruction for quantum tomography. *Phys. Rev. A* **95**, 062336 (2017).
39. Gonçalves, D., Gomes-Ruggiero, M. & Lavor, C. A projected gradient method for optimization over density matrices. *Optim. Methods Softw.* **31**(2), 328–341 (2016).
40. Bolduc, E., Knee, G., Gauger, E. & Leach, J. Projected gradient descent algorithms for quantum state tomography. *NPJ Quantum Inf.* **3**(1), 44 (2017).
41. Nesterov, Y. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Sov. Math. Dokl.* **27**, 372–376 (1983).
42. Kyrillidis, A. & Cevher, V. Matrix recipes for hard thresholding methods. *J. Mathematic Imaging Vision.* **48**(2), 235–265 (2014).
43. Riofrío, C. et al. Experimental quantum compressed sensing for a seven-qubit system. *Nat. Commun.* **8**, 15305 (2017).
44. Christandl, M. & Renner, R. Reliable quantum state tomography. *Phys. Rev. Lett.* **109**, 120403 (2012).
45. Shang, J., Ng, H. K., Sehrawat, A., Li, X. & Englert, B.-G. Optimal error regions for quantum state estimation. *New J. Phys.* **15**, 123026 (2013).
46. Hou, Z. et al. Full reconstruction of a 14-qubit state within four hours. *New J. Phys.* **18**(8), 083036 (2016).