Faster Stochastic First-Order Method for Maximum-Likelihood Quantum State Tomography

Chung-En Tsai[1], Hao-Chung Cheng[2,3,4,5], and Yen-Huan Li[1,3,5]

[1]Department of Computer Science and Information Engineering,
National Taiwan University
[2]Department of Electrical Engineering and Graduate Institute of Communication Engineering, National Taiwan University
[3]Department of Mathematics, National Taiwan University
[4]Hon Hai (Foxconn) Quantum Computing Centre
[5]Center for Quantum Science and Engineering, National Taiwan University

Abstract

In maximum-likelihood quantum state tomography, both the sample size and dimension grow exponentially with the number of qubits. It is therefore desirable to develop a stochastic first-order method, just like stochastic gradient descent for modern machine learning, to compute the maximum-likelihood estimate. To this end, we propose an algorithm called stochastic mirror descent with the Burg entropy. Its expected optimization error vanishes at a $O(\sqrt{(1/t)d\log t})$ rate, where d and t denote the dimension and number of iterations, respectively. Its per-iteration time complexity is $O(d^3)$, independent of the sample size. To the best of our knowledge, this is currently the computationally fastest stochastic first-order method for maximum-likelihood quantum state tomography.

1 Introduction

In maximum-likelihood quantum state tomography [1], the estimate of the unknown quantum state is given by

$$\hat{\rho} \in \operatorname*{arg\,min}_{\rho \in \mathcal{D}} f(\rho), \quad \text{where } f(\rho) \coloneqq \frac{1}{n} \sum_{i=1}^{n} -\log \operatorname{tr}(A_i \rho), \tag{1}$$

where A_i are Hermitian positive semi-definite matrices determined by the measurement outcomes and $\mathcal{D}$ denotes the set of density matrices, i.e.,

$$\mathcal{D} \coloneqq \left\{ \rho \in \mathbb{C}^{d\times d} : \rho^* = \rho, \rho \geq 0, \operatorname{tr} \rho = 1 \right\}.$$

We call n the *sample size.*

The optimization problem (1) is apparently convex. Solving it rigorously and efficiently, however, is surprisingly difficult. The first challenge is the computational scalability with respect to the sample size n. Consider computing $\hat{\rho}$ by projected gradient descent, perhaps the simplest convex optimization algorithm [2]. Projected gradient descent iterates as

$$\rho_{t+1} = \Pi_{\mathcal{D}} \left(\rho_t - \eta \nabla f(\rho_t) \right), \quad \forall t \in \mathbb{N},$$

given some initial iterate $\rho_1 \in \mathcal{D}$ and step size $\eta > 0$, where $\Pi_{\mathcal{D}}$ denotes projection onto $\mathcal{D}$ with respect to the Frobenius norm. In each iteration, computing the projection onto $\mathcal{D}$ takes time $O(d^3)$ [3]; computing the gradient,

$$\nabla f(\rho_t) = \frac{1}{n} \sum_{i=1}^{n} \frac{-A_i}{\operatorname{tr}(A_i \rho_t)},$$

takes time $O(nd^\omega)$, where ω, $2 \leq \omega < 2.373$ is the matrix multiplication exponent [4]; the overall per-iteration time is hence $O(d^3 + nd^\omega)$. It has been shown that any quantum state tomography scheme with incoherent measurements, including maximum-likelihood quantum state tomography, requires $n = \Omega(d^3/\varepsilon^2)$ to achieve an estimation error smaller than ε in the trace norm [5]. The dependence of the per-iteration time on the sample size n becomes the main computational bottleneck.

Another challenge lies in the fact that rigorously speaking, projected gradient descent is not directly applicable to solving (1). The reason is two-fold. First, the function f is neither Lipschitz nor Lipschitz gradient, violating standard assumptions in the literature (see, e.g., the textbook by Nesterov [6]). It is unclear how to choose the step size η to guarantee convergence of the projected gradient descent. Newton's method is guaranteed to converge [6]. Regarding the exponentially growing dimension, however, computations involving the Hessian are computationally too expensive. Second, in standard implementations of maximum-likelihood quantum state tomography, the matrices A_i are low rank, either of rank $d/2$ with Pauli measurements or of rank 1 with Pauli basis measurements. Projection onto $\mathcal{D}$ typically results in low-rank matrices [3]. The trace of the product of two low-rank matrices can be exactly zero. It can happen that $\operatorname{tr}(A_i \rho_t) = 0$ for some A_i and some iterate ρ_t of projected gradient descent; then, $f(\rho_t)$ and $\nabla f(\rho_t)$ is undefined and projected gradient descent is forced to "stall." We refer to the discussions by Knee et al. [7] and You et al. [8] for the details.

Previously, we developed an algorithm, Stochastic Q-Soft-Bayes, to address the two challenges above [9].

- It only involves one randomly selected A_i, like stochastic gradient descent in modern machine learning, in each iteration. Therefore, its per-iteration time is $O(d^3 + d^\omega)$, independent of the sample size n[1].
- It always generates full-ranked iterates, thereby avoiding the "stalling" issue[2].
- It has a rigorous non-asymptotic convergence guarantee. Its expected optimization error vanishes at a $O(\sqrt{(1/t) d \log d})$ rate.

In this paper, we propose a completely different algorithm, stochastic mirror descent with the Burg entropy. The algorithm shares the same three merits above, except that the convergence rate is $O(\sqrt{(1/t) d \log t})$, slightly worse than Stochastic Q-Soft-Bayes when $t \gg d$. Nevertheless, its per-iteration time is significantly shorter. Each iteration of Stochastic Q-Soft-Bayes requires computing two matrix logarithms and one matrix exponential, whereas each iteration of stochastic mirror descent with the Burg entropy only requires computing one eigendecomposition. Though for both algorithms, the per-iteration time is $O(d^3)$ in theory, the difference in per-iteration time is significant in practice. Numerical results verify the competitiveness of the proposed algorithm compared to Stochastic Q-Soft-Bayes. The results also show the stochastic methods have some space for improvement compared to non-stochastic ones.

2 Related Work

There are several approaches to quantum state tomography [10, 1, 11, 12, 13, 14, 15, 16]. This paper focuses on the maximum-likelihood approach. In particular, this papers focuses on how to compute the maximum-likelihood estimate efficiently. Notice that the statistical performance—the statistical estimation error–is already determined by the formulation of the maximum-likelihood estimator. The interested reader is referred to, e.g., Scholten and Blume-Kohout [17] for a statistical analysis of the maximum-likelihood estimator.

As discussed in Section 1, textbook convex optimization algorithms do not direct apply to computing the maximum-likelihood estimate. There are some advanced algorithms that are applicable, such as diluted $R\rho R$ [18], SCOPT [19], NoLips [20], entropic mirror descent with line search [21], and several variants of the Frank-Wolfe method [22, 23, 24]. Among them, diluted $R\rho R$ and entropic mirror descent with line search are only guaranteed to converge asymptotically; SCOPT only has a local convergence rate guarantee. Hence, their computational complexities are unclear. NoLips and the Frank-Wolfe methods proposed by Dvurechensky et al. [22] and Carderera et al. [23] all converge at a $O(1/t)$ rate; the dependence of their convergence rates on

[1] The term d^ω is irrelevant as $\omega < 3$. We include it simply for the reader to easily check how the time complexity is evaluated.

[2] Notice that this does not prevent the algorithm from converging to a singular minimizer.

the problem parameters, such as the dimension and sample size, are unclear. Hence, it is unclear whether they scale with the dimension and sample size or not. The Frank-Wolfe method proposed by Zhao and Freund [24] converges at a $O(n/t)$ rate, but each iteration of the algorithm involves Hessian computations, computationally expensive under the quantum setup.

For computing a quantum state estimate by stochastic first-order methods, we are only aware of two existing results. Youssry et al. [25] proposed stochastic entropic mirror descent and proved its asymptotic convergence for the least-squares regression approach to quantum state tomography. Lin et al. [9] proposed Stochastic Q-Soft-Bayes for the maximum-likelihood approach and derived a non-asymptotic convergence guarantee, as discussed in Section 1. This paper aims to improve on the algorithm by Lin et al. [9] in terms of the computational efficiency.

There are a few online learning formulations of estimating quantum states. Yang et al. [26] studied several online convex optimization algorithms for Lipschitz losses. Unfortunately, as mentioned in Section 1, the logarithmic loss in (1) is not Lipschitz. Zimmert et al. [27] developed an algorithm for online learning quantum states with the logarithmic loss. Their algorithm, after an online-to-batch conversion (see the discussion in Section 3.1), also solves (1). The algorithm, however, involves $t-1$ data points in the t-th iteration and adopts Newton's method to compute each iterate. Therefore, the algorithm does not scale with the sample size and dimension.

A closely related but essentially different problem is shadow tomography [28]. Shadow tomography aims to estimate the probability distribution of measurement outcomes instead of the whole density matrix. Therefore, it has been proven that shadow tomography requires a much smaller sample size than state tomography. Shadow tomography can be implemented by an online learning algorithm [29]. Though shadow tomography is efficient in sample size, we notice that the online shadow tomography algorithm still has per-iteration time $O(d^3)$ as our algorithm.

3 Algorithm and Convergence Guarantee

3.1 Stochastic mirror descent with the Burg entropy

The proposed algorithm, stochastic mirror descent with the Burg entropy, is shown in Algorithm 1. In the algorithm,

$$\nabla f_t(\overline{\rho}_t) = \frac{-A_{i_t}}{\operatorname{tr}(A_{i_t}\overline{\rho}_t)},$$

and D_h denotes the Bregman divergence with respect to the Burg entropy, which we denote by h, i.e.,

$$\begin{aligned} &D_h(\rho,\sigma) \coloneqq h(\rho) - h(\sigma) - \operatorname{tr}(\nabla h(\sigma)(\rho-\sigma)), \\ &h(\rho) \coloneqq -\log\det\rho, \\ &\nabla h(\rho) = -\rho^{-1}. \end{aligned}$$

Algorithm 1 Stochastic mirror descent with the Burg entropy.

```
Input: $\eta > 0$.
$\rho_1 = I/d$.
for all $t \in \mathbb{N}$ do
    $\overline{\rho}_t \leftarrow (1/t)\sum_{\tau=1}^{t} \rho_\tau$.
    Output $\overline{\rho}_t$.
    Sample an index $i_t \in \{1,\dots,n\}$ uniformly and independently of the past.
    $f_t(\rho) \coloneqq -\log\operatorname{tr}(A_{i_t}\rho)$.
    $\rho_{t+1} \leftarrow \arg\min_{\rho\in\mathcal{D}} \eta\operatorname{tr}(\nabla f_t(\overline{\rho}_t)(\rho-\rho_t)) + D_h(\rho,\rho_t)$.
end for
```

The proposed algorithm is based on online mirror descent with the Burg entropy, a recent algorithm we developed for online learning a quantum state with the logarithmic loss. By the logarithmic loss, we mean loss functions of the form $\rho \mapsto -\log\operatorname{tr}(A\rho)$ for some Hermitian positive semi-definite matrix A. After an online-to-batch conversion, online mirror descent with the Burg entropy can be transformed to an algorithm minimizing an expectation of the logarithmic loss [30, 31]. Notice that the function f we want to minimize is exactly the expected logarithmic loss

Algorithm 2 Efficient computation of ρ_{t+1}.

```
Input: $\eta > 0, \varepsilon > 0, A_{i_t}, \overline{\rho}_t, \rho_t, \rho_t^{-1}$.
$\nabla \leftarrow -A_{i_t} / \operatorname{tr}(A_{i_t} \overline{\rho}_t)$.
$(\lambda_1, \dots, \lambda_d), U \leftarrow \texttt{eig}(\eta \nabla + \rho_t^{-1})$.
$\phi(\theta) := \theta - \sum_{i=1}^d \log(\theta + \lambda_i)$.
$\theta \leftarrow 1 - \min_{i \in [d]} \lambda_i$.
while $|\phi'(\theta)| / \sqrt{\phi''(\theta)} \geq \varepsilon$ do
    $\theta \leftarrow \theta - \phi'(\theta) / \phi''(\theta)$.
end while
for all $i \in [d]$ do
    $\lambda_i' \leftarrow (\theta + \lambda_i)^{-1}$.
end for
$\Lambda' \leftarrow \operatorname{diag}(\lambda_1', \dots, \lambda_d')$
$\rho_{t+1} \leftarrow U \Lambda U^*$.
$\rho_{t+1}^{-1} \leftarrow U \Lambda^{-1} U^*$.
return $(\rho_{t+1}, \rho_{t+1}^{-1})$.
```

with respect to the uniform distribution on $\{ A_1, \dots, A_n \}$, or the empirical distribution on the data, so online mirror descent with the Burg entropy followed by an online-to-batch conversion readily applies. There are several online-to-batch methods. Algorithm 1 adopts the "anytime online-to-batch" procedure by Cutkosky [31], as its empirical performance seems to be better.

Notice that each iteration of Algorithm 1 only involves one randomly selected A_{i_t}, instead of $A_1, \dots, A_n$. Therefore, its per-iteration time is independent of the sample size n.

3.2 Efficient Implementation

The seventh line in Algorithm 1 does not have a closed-form expression, but can be efficiently computed. Following the discussion by Kotłowski and Neu [32], it suffices to:

1. compute the eigendecomposition of $\eta \nabla f_t(\overline{\rho}_t) + \rho_t^{-1}$ and get a vector λ of eigenvalues and a matrix U consisting of eigenvectors;
2. project the vector λ onto the the probability simplex, the set of entrywise-nonnegative vectors of unit 1-norm, with respect to the logarithmic barrier and get a vector λ'; and
3. compute the matrix $U \operatorname{diag}(\lambda') U^*$ as ρ_{t+1}.

The projection with respect to the logarithmic barrier can be computed by Newton's method on the real line $\mathbb{R}$, which takes very short time in practice. We detail the steps above in Algorithm 2 for the reader's convenience. In Algorithm 2, $\texttt{eig}(A)$ is a procedure that computes the eigendecomposition of A. The eigenvectors are returned in a matrix U. The parameter ε specifies the desired error achieved by Newton's method.

The computational bottleneck in Algorithm 2 is the eigendecomposition. It is easily checked, then, that the per-iteration time of Algorithm 1 is $O(d^3)$.

3.3 Convergence Guarantee

The following non-asymptotic convergence guarantee of Algorithm 1 follows directly from the regret bound of online mirror descent with the Burg entropy in [33] and the analysis of anytime online-to-batch conversion [31].

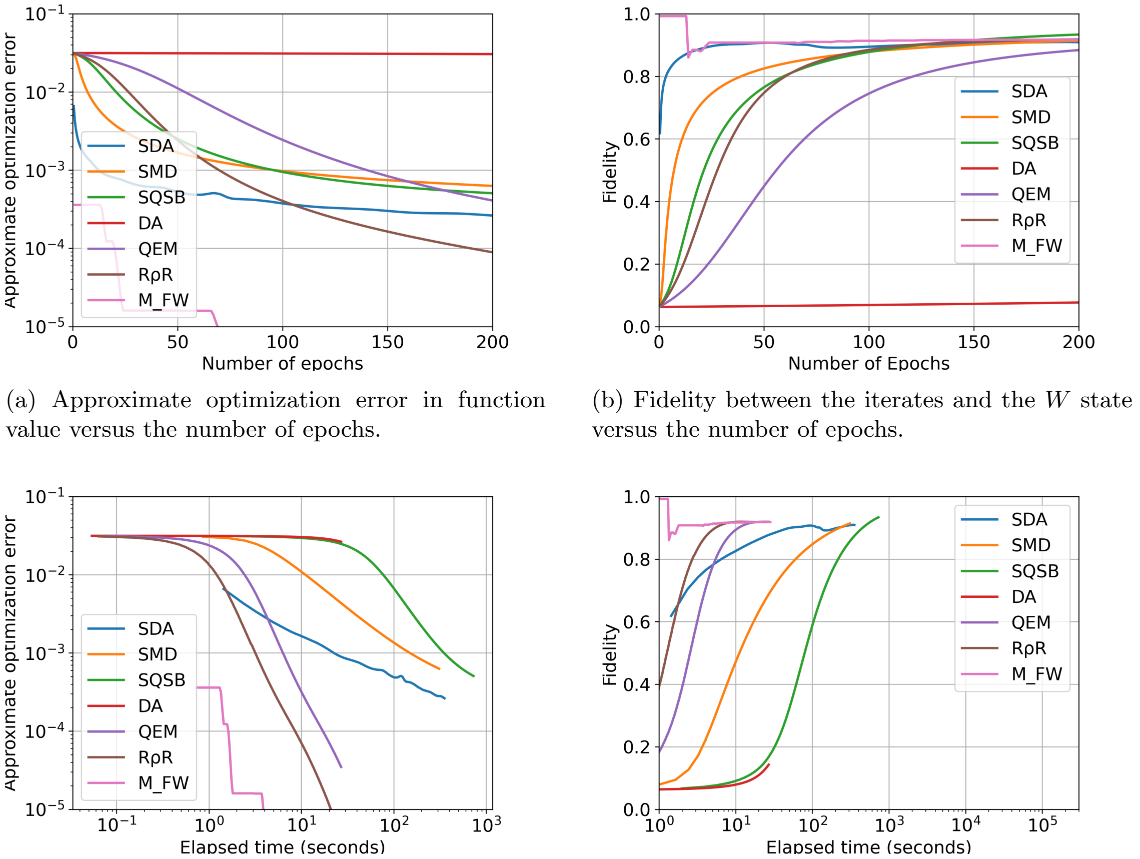
Theorem 1. *Fix $T \in \mathbb{N}$. Let $\{\overline{\rho}_t\}_{t \in \mathbb{N}}$ be the iterates generated by Algorithm 1 with*

$$\eta = \frac{\sqrt{d \log T}}{\sqrt{T} + \sqrt{d \log T}}.$$

Then it holds that

$$\mathsf{E}\left[f(\overline{\rho}_T) - \min_{\rho \in \mathcal{D}} f(\rho) \right] \leq 2 \sqrt{\frac{d \log T}{T}} + \frac{d \log T}{T},$$

where the expectation is taken with respect to the random choices of A_{i_t} in Algorithm 1.



(a) Approximate optimization error in function value versus the number of epochs.

(b) Fidelity between the iterates and the W state versus the number of epochs.

(c) Approximate optimization error in function value versus the elapsed time.

(d) Fidelity between the iterates and the W state versus the elapsed time.

Figure 1: Comparison of the performances of Algorithm 1, stochastic Q-Soft-Bayes (SQSB), monotonous Frank-Wolfe (FW), and $R\rho R$.

4 Numerical Results

This section compares the performances of Algorithm 1, stochastic Q-Soft-Bayes, monotonous Frank-Wolfe, and $R\rho R$ [34, 35] on synthetic data. Stochastic Q-Soft-Bayes has been discussed in previous sections. Monotonous Frank-Wolfe and $R\rho R$ are not stochastic methods; each iteration of the algorithms require processing the whole data-set. We include the two algorithms for comparison as they are easy to implement. Monotonous Frank-Wolfe is one of the variants of Frank-Wolfe discussed in Section 2. Its convergence rate has unclear dependence on the dimension and sample size. $R\rho R$ is a predecessor of diluted $R\rho R$, discussed in Section 2. $R\rho R$ typically converges fast in practice, but does not work on carefully designed data-sets. Notice the comparison with monotonous Frank-Wolfe and $R\rho R$ is rigorously speaking unfair, as the two algorithms lack clear non-asymptotic convergence guarantees.

The numerical results are shown in Figure 1. The data-set is synthetic. There are six qubits, so $d = 2^6 = 64$. The unknown density matrix to be estimated is the W-state. The sample size n is set to $4^6 \times 100 = 409600$. Each data point corresponds to the measurement outcome of a randomly chosen Pauli observable.

In Figure 1, one epoch means one full pass of the data-set. One iteration of Algorithm 1 and Stochastic Q-Soft-Bayes corresponds to $1/n$ epoch, whereas one iteration of $R\rho R$ and monotonous Frank-Wolfe corresponds to one epoch. The exact minimizer in (1), i.e., $\hat{\rho}$, is not accessible. We run all algorithms for 200 epochs and set the smallest values of f achieved as $\widehat{f^\star}$. Then, the approximate optimization error is the difference between the value of f achieved at an iterate and $\widehat{f^\star}$. The elapsed time is recorded on a machine with an Intel Xeon Gold 5218 CPU of 2.30GHz and 131621512kB memory. The four algorithms are implemented with the Julia language with the Intel Math Kernel Library (MKL). The number of threads for BLAS is set to 8. It should be mentioned that the empirical speed depends highly on the implementations.

Regarding the number of epochs, Algorithm 1 converges slightly faster than Stochastic Q-Soft-Bayes and much faster than $R\rho R$. Monotonous Frank-Wolfe is significantly faster than the others. An explanation given by Lin et al. [9] is that Frank-Wolfe prefers low-rank iterates, which matches the structure of the W-state, a rank-1 density matrix. Algorithm 1 converges faster than Stochastic Q-Soft-Bayes in the beginning but the gap vanishes in the end. This may be explained by the fact that the expected optimization error of Algorithm 1 is $O(\sqrt{d \log T}/\sqrt{T})$, which is slightly worse than $O(\sqrt{d \log d}/\sqrt{T})$ of stochastic Q-Soft-Bayes when $T \gg d$.

Regarding the elapsed time, Algorithm 1 is about 2.3-times faster than Stochastic Q-Soft-Bayes (70,227 seconds vs. 160,849 seconds), although their per-iteration time are the same in the big-O notation. Recall that each iteration of Stochastic Q-Soft-Bayes involves two matrix logarithms and one matrix exponential; either is slower than the eigendecomposition needed in Algorithm 1. This explains why Algorithm 1 is faster than stochastic Q-Soft-Bayes in practice.

The two stochastic methods are still slower than $R\rho R$ and monotonous Frank-Wolfe in terms of the elapsed time, showing some space for improvement for stochastic methods. Notice that both the number of iterations and per-iteration time affect the real elapsed time. $R\rho R$ and monotonous Frank-Wolfe actually have significantly shorter per-iteration time; $O(d^\omega)$ for the former and $O(d^2)$ for the latter. Nevertheless, it seems difficult to characterize the required number of iterations, because $R\rho R$ and monotonous Frank-Wolfe lack clear non-asymptotic convergence guarantees.

Acknowledgments

C.-E. Tsai and Y.-H. Li are supported by the Young Scholar Fellowship (Einstein Program) of the National Science and Technology Council of Taiwan under grant numbers MOST 108-2636-E-002-014, MOST 109-2636-E-002-025, MOST 110-2636-E-002-012 and MOST 111-2636-E-002-019 and by the research project "Pioneering Research in Forefront Quantum Computing, Learning and Engineering" of National Taiwan University under grant number NTU-CC-111L894606.

H.-C. Cheng is supported by the Young Scholar Fellowship (Einstein Program) of the National Science and Technology Council (NSTC) in Taiwan (R.O.C.) under Grant MOST 111-2636-E-002-026, Grand MOST 111-2119-M-007-006, Grant MOST 111-2119-M-001-004, and is supported by the Yushan Young Scholar Program of the Ministry of Education in Taiwan (R.O.C.) under Grant NTU-111V1904-3, and Grant NTU-111L3401, and by the research project "Pioneering Research in Forefront Quantum Computing, Learning and Engineering" of National Taiwan University under Grant No. NTU-CC-111L894605."

References

[1] Z. Hradil. Quantum-state estimation. *Phys. Rev. A*, 55, 1997.

[2] Eliot Bolduc, George C. Knee, Erik M. Gauger, and Jonathan Leach. Projected gradient descent algorithms for quantum state tomography. *npj Quantum Inf.*, 3, 2017.

[3] Anastasios Kyrillidis, Stephen Becker, Volkan Cevher, and Christoph Koch. Sparse projections onto the simplex. In *Proc. 30th Int. Conf. Machine Learning*, pages 235–243, 2013.

[4] Josh Alman and Vriginia Vassilevska Williams. A refined laser method and faster matrix multiplication. In *Proc. 2021 ACM-SIAM Symp. Discrete Algorithms (SODA)*, 2021.

[5] Sitan Chen, Brice Huang, Jerry Li, Allen Liu, and Mark Sellke. Tight Bounds for State Tomography with Incoherent Measurements. 2022. arXiv:2206.05265.

[6] Yurii Nesterov. *Lectures on Convex Optimization.* Springer, Cham, CH, second edition, 2018.

[7] George C. Knee, Eliot Bolduc, Jonathan Leach, and Erik M. Gauger. Quantum process tomography via completely positive and trace-preserving projection. *Phys. Rev. A*, 2018.

[8] Jun-Kai You, Hao-Chung Cheng, and Yen-Huan Li. Minimizing quantum Rényi divergences via mirror descent with Polyak step size. In *IEEE Int. Symp. Information Theory (ISIT)*, 2022.

[9] Chien-Ming Lin, Yu-Ming Hsu, and Yen-Huan Li. An online algorithm for maximum-likelihood quantum state tomography. 2021. arXiv:2012.15498.

[10] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information.* Cambridge Univ. Press, Cambridge, UK, 2010.

[11] David Gross, Yi-Kai Liu, Steven T. Flammia, Stephen Becker, and Jens Eisert. Quantum state tomography via compressed sensing. *Phys. Rev. Lett.*, 105, 2010.

[12] Robin Blume-Kohout. Hedged maximum likelihood quantum state estimation. *Phys. Rev. Lett.*, 105, 2010.

[13] Ryan O'Donnell and John Wright. Efficient quantum tomography. In *Proc. 48th Annu. ACM Symp. Theory of Computing*, pages 899–912, 2016.

[14] Jeongwan Haah, Aram W. Harrow, Zhengfeng Ji, Xiaodi Wu, and Nengkun Yu. Sample-optimal tomography of quantum states. *IEEE Trans. Inf. Theory*, 63(9):5628–5641, 2017.

[15] Giacomo Torlai, Guglielmo Mazzola, Juan Carrasquilla, Matthias Troyer, Roger Melko, and Giuseppe Carleo. Neural-network quantum state tomography. *Nat. Phys.*, 14:447–450, 2018.

[16] M. Guţă, J. Kahn, R. Kueng, and J. A. Tropp. Fast state tomography with optimal error bounds. *J. Phys. A: Math. Theor.*, 53, 2020.

[17] Travis L. Scholten and Robin Blume-Kohout. Behavior of maximum likelihood in quantum state tomography. *New J. Phys.*, 20, 2018.

[18] Jaroslav Řeháček, Zdeněk Hradil, E. Knill, and A. I. Lvovsky. Diluted maximum-likelihood algorithm for quantum tomography. *Phys. Rev. A*, 75(4):042108, 2007.

[19] Quoc Tran-Dinh, Anastasios Kyrillidis, and Volkan Cevher. Composite Self-Concordant Minimization. *Journal of Machine Learning Research*, 16(12):371–416, 2015.

[20] Heinz H. Bauschke, Jérôme Bolte, and Marc Teboulle. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. *Math. Oper. Res.*, 42(2): 330–348, 2017.

[21] Yen-Huan Li and Volkan Cevher. Convergence of the exponentiated gradient method with Armijo line search. *J. Optim. Theory Appl.*, 181(2):588–607, 2019.

[22] Pavel Dvurechensky, Petr Ostroukhov, Kamil Safin, Shimrit Shtern, and Mathias Staudigl. Self-concordant analysis of Frank-Wolfe algorithms. In *Proc. 37th Int. Conf. Machine Learning*, 2020.

[23] Alejandro Carderera, Mathieu Besançon, and Sebastian Pokutta. Simple steps are all you need: Frank-Wolfe and generalized self-concordant functions. In *Adv. Neural Information Processing Systems 34*, volume 34, pages 5390–5401, 2021.

[24] Renbo Zhao and Robert M. Freund. Analysis of the Frank–Wolfe method for convex composite optimization involving a logarithmically-homogeneous barrier. *Math. Program.*, 2022.

[25] Akram Youssry, Christopher Ferrie, and Marco Tomamichel. Efficient online quantum state estimation using a matrix-exponentiated gradient method. *New J. of Phys.*, 21(3):033006, 2019.

[26] Feidiao Yang, Jiaqing Jiang, Jialin Zhang, and Xiaoming Sun. Revisiting online quantum state learning. In *Proc. AAAI Conf. Artificial Intelligence*, 2020.

[27] Julian Zimmert, Naman Agarwal, and Satyen Kale. Pushing the efficiency-regret Pareto frontier for online learning of portfolios and quantum states. In *Proc. 35th Conf. Learning Theory*, pages 182–226, 2022.

[28] Scott Aaronson. Shadow tomography of quantum states. *SIAM J. Comput.*, 49(5):STOC18–368–STOC18–394, 2020.

[29] Scott Aaronson, Xinyi Chen, Elad Hazan, Satyen Kale, and Ashwin Nayak. Online learning of quantum states. In *Adv. Neural Information Processing Systems 31*, 2018.

[30] N. Cesa-Bianchi, A. Conconi, and C. Gentile. On the generalization ability of on-line learning algorithms. *IEEE Trans. Inf. Theory*, 50(9):2050–2057, 2004.

[31] Ashok Cutkosky. Anytime Online-to-Batch, Optimism and Acceleration. In *Proc. 36th Int. Conf. on Mach. Learn.*, volume 97, pages 1446–1454, 2019.

[32] Wojciech Kotłowski and Gergely Neu. Bandit principal component analysis. In *Proc. 32nd Conf. Learning Theory*, pages 1994–2024, 2019.

[33] Chung-En Tsai, Hao-Chung Cheng, and Yen-Huan Li. Online Self-Concordant and Relatively Smooth Minimization, With Applications to Online Portfolio Selection and Learning Quantum States. 2022. arXiv:2210:00997 [stat.ML].

[34] A. I. Lvovsky. Iterative maximum-likelihood reconstruction in quantum homodyne tomography. *J. of Opt. B: Quantum and Semiclass. Opt.*, 6(6):S556, 2004.

[35] G. Molina-Terriza, A. Vaziri, J. Řeháček, Z. Hradil, and A. Zeilinger. Triggered qutrits for quantum communication protocols. *Phys. Rev. Lett.*, 92(16), 2004.