Efficient Quantum State Sample Tomography with Basis-Dependent Neural Networks

Alistair W. R. Smith,[1,*] Johnnie Gray,[1,2] and M. S. Kim[1]

[1] *QOLS, Blackett Laboratory, Imperial College, London SW7 2AZ, United Kingdom*

[2] *Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, California 91125, USA*

(Received 18 August 2020; accepted 1 June 2021; published 28 June 2021)

We use a metalearning neural-network approach to analyze data from a measured quantum state. Once our neural network has been trained, it can be used to efficiently sample measurements of the state in measurement bases not contained in the training data. These samples can be used to calculate expectation values and other useful quantities. We refer to this process as "state sample tomography." We encode the state's measurement outcome distributions using an efficiently parameterized generative neural network. This allows each stage in the tomography process to be performed efficiently even for large systems. Our scheme is demonstrated on recent IBM Quantum devices, producing a model for a six-qubit state's measurement outcomes with a predictive accuracy (classical fidelity) greater than 95% for all test cases using only 100 random measurement settings as opposed to the 729 settings required for standard full tomography using local measurements. This reduction in the required number of measurements scales favorably, with training data in 200 measurement settings, yielding a predictive accuracy greater than 92% for a ten-qubit state where 59 049 settings are typically required for full local measurement-based quantum state tomography. A reduction in the number of measurements by a factor, in this case, of almost 600 could allow for estimations of expectation values and state fidelities in practicable times on current quantum devices.

DOI: 10.1103/PRXQuantum.2.020348

I. INTRODUCTION

There are large practical hurdles to overcome when attempting to perform quantum state tomography (QST) on noisy quantum devices. The extremely rapid scaling in the number of measurements that are required to completely specify a state, even at relatively small system sizes, makes full QST infeasible [1]. Full, direct QST on n qubits requires the estimation of $4^n - 1$ linearly independent Stokes parameters to specify a completely general quantum state and is typically performed using local projective measurements onto the eigenstates of all 3^n nonidentity-containing Pauli strings [2]. While this set of measurements is overcomplete (containing 6^n quantities in total), this informationally complete protocol allows the state to be specified with the smallest number of measurement settings possible when using local measurements, and so is the most effective way of implementing full tomography on a general quantum state [2]. More efficient protocols exist [3] that do not involve collection overcomplete sets of measurement data; however, these invariably require measurements onto entangled states [4]. While these entangled measurements can be indirectly performed on-device through the application of a nonlocal unitary before the local measurement [5], the imperfect entangling gates that are required introduce significant amounts of noise. As this noise changes the on-device state, measurements taken in different entangled bases will not properly correspond to the original desired state, and so these measurement-efficient schemes are not typically implemented on current devices. These measurements must each be repeated on many copies of the state to build up sufficient statistics to accurately estimate the Stokes parameters. This is usually followed by a maximum likelihood estimation (MLE) procedure [2,6–8] to ensure that the quantum state that is inferred is valid. As the MLE is constructing a density matrix whose dimension scales exponentially with the system size, this step quickly becomes prohibitively expensive.

The noise on present-day quantum devices inevitably leads to errors in running the quantum circuits that produce

*alistair.smith18@imperial.ac.uk

the states. Even if we knew that an ideal preparation of the state could be reconstructed with a more limited set of measurement bases, the complexity and unpredictability of these noise processes make it difficult to know *a priori* which limited measurement set would be most informative. To illustrate the scale of this issue, we can consider the time cost of performing full QST on a ten-qubit state. In our experiments we find that a single circuit on an IBM Quantum device takes of the order of 1 ms to calibrate and run. Running each of the required $3^{10} = 59\,049$ circuits 8192 times (the maximum allowed shot count per circuit) to build up reasonably accurate measurement statistics, it would take around 130 h to collect the data needed for full QST. These devices are frequently recalibrated and their errors drift over time, meaning that the states measured at the end of this process may vary significantly from those prepared at the start [9]. On devices that are in high demand, long queue times can compound this problem even further, making full QST completely impractical for systems of this size.

Many physically relevant states have some underlying structure (often due to symmetries in the system), meaning that an n-qubit state can be described with fewer parameters than the worst case $4^n - 1$ [10–13]. Numerous tomographic methods have been proposed that exploit this structure in order to perform tomography more efficiently, describing the state with fewer parameters that in turn require fewer measurements to learn [14–17]. These concise descriptions typically provide more efficient ways of manipulating the description of the state than is possible with an explicit density matrix. Recent examples of these include machine-learning (ML) inspired approaches that attempt to take advantage of the ability of neural networks to efficiently represent and learn complicated probability distributions. These approaches have been shown to perform quantum state tomography very effectively for systems of small to intermediate size, requiring fewer measurements and being more computationally efficient than standard MLE-based techniques [18]. An early example of such a scheme was proposed by Torlai [14] and is based on the "neural-network quantum state" representation introduced by Carleo and Troyer [10]. It involves representing a pure quantum state in terms of two restricted Boltzmann machines (RBM; a popular generative neural network architecture). This scheme has many useful properties; for many states, it can provide a very concise representation of the state (although, to calculate the state exactly, an exponentially costly partition function must be found), it can represent states possessing long-range entanglement, the optimization of the ansatz is very computationally efficient for suitable states (using Hinton's contrastive divergence algorithm [19]), and as a generative neural-network, it is simple and computationally efficient to draw samples from the state in the computational basis [14,20,21]. By efficient what is meant here is that the computational cost of each step in these processes scales polynomially with the number of subsystems or qubits.

Neural network (NN) approaches that attempt to produce an explicit description of quantum states in terms of density matrices or pure state vectors run into difficulties for two reasons. First, density matrices are unwieldy objects, their size scaling exponentially in the number of subsystems. This means that using them to calculate quantities, either for optimizing the ansatz or analyzing the inferred state, involves sums of exponentially many (potentially interfering) terms. If one is able to draw samples from the represented state in the relevant bases then these quantities can be efficiently calculated and this is the route that neural network approaches tend to take [14,17,22]. However, this then leads to the second issue; density matrix and state vector descriptions of quantum states involve complex-valued components, whereas neural networks are typically designed to encode real-valued distributions [23]. These complex values are required to determine measurement distributions in different bases. If one tries to account for this by using neural networks with complex-valued parameters (as in with the neural-network quantum state representation given in Ref. [10]) or treating the moduli and phase of state components separately [14], then it becomes difficult to train the network efficiently and draw samples in bases that differ greatly from the computational basis (the cost of doing so scales exponentially with the number of qubits).

Rather than trying to reconcile this fundamental difference between neural networks and quantum states, we attempt to sidestep the issue, along with the scaling problems that can make explicit density matrices impractical to use. We skip the density matrix reconstruction stage and instead build a model of how the measurement outcome distributions vary with the choice of measurement basis. The model can then make predictions of the measurement outcome distributions in the arbitrary local basis (we refer to this as "distribution prediction"). Calculations using the full distribution of outcome probabilities quickly become intractable for large systems (there are 2^n outcome probabilities for n qubits), making the full explicit outcome distribution of limited utility. For many parameterized probability distributions, a regularization step (calculation of a partition function) is needed to obtain the exact probabilities—for large systems, this step becomes intractable. To avoid the difficulties in using full explicit outcome distributions, we use a model in which this distribution is implicitly defined in terms of an efficient number of parameters; the unnormalized probabilities of individual outcomes can be efficiently calculated, but the normalization requires an inefficient partition function calculation. While explicit probabilities are not easily accessible, as in the approach taken by Torlai [14], we can draw samples from the implicitly represented measurement outcome distributions

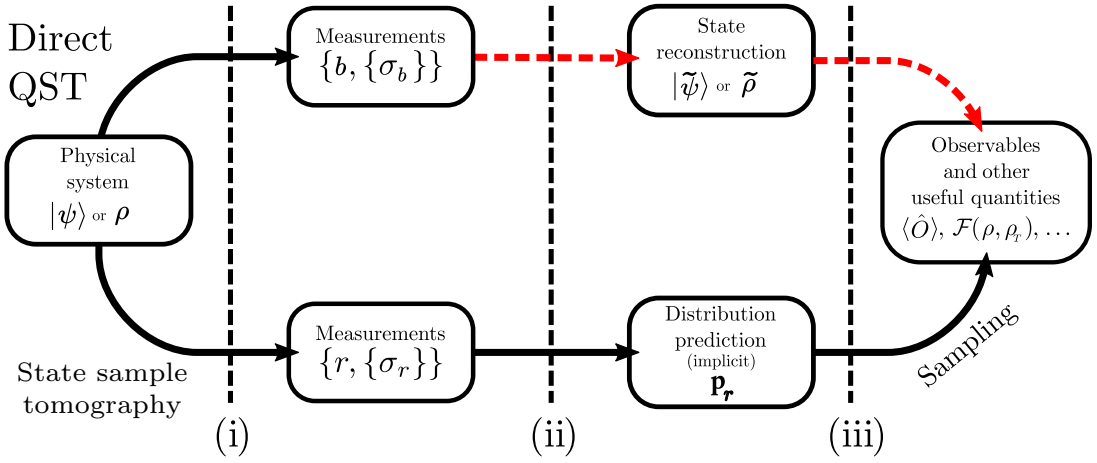

FIG. 1. Illustration of direct QST versus SST. Here we show the differences between direct QST, in which the density matrix is inferred, and state sample tomography where we build a model to implicitly predict measurement outcome distributions and then sample from them efficiently. The end goal in both cases is the calculation of quantities such as observables $\langle\hat{O}\rangle$ and state fidelities $\mathcal{F}(\rho, \rho_T)$. Lines in red show stages that become intractable for large systems when using direct QST. (i) Measurements $\{\sigma_b\}$ ($\{\sigma_r\}$) are taken of the quantum state in various bases $\{b\}$ ($\{r\}$)—the choice of bases depends on the approach taken. Whether direct QST or SST is inherently more expensive in the required number of measurements is beyond the scope of this work but, in general, depends on the exact methods used and the state in question. (ii) Training (or reconstruction) stage: in SST a model is trained to map measurement settings r to outcome distributions $\mathbf{p}_r$; the BDRBM model that we present performs this stage efficiently provided each local measurement distribution can be tractably approximated by an RBM. Direct QST in this stage involves optimization over an exponentially large number of, potentially complex-valued, parameters to produce an exponentially large density matrix. (iii) Calculation stage: once the tomography has produced a model of either the state or its measurement distributions, this is used to calculate some desired quantities, which in the setting presented here are those that can be calculated efficiently from measurement results in local bases. SST allows these to be efficiently calculated by sampling from the predicted measurement distributions. Direct QST requires manipulation of large density matrices that are difficult to manipulate and sample from.

in each basis efficiently, and then use these to calculate quantities of interest—we refer to this as "sample prediction." We call the overall process of implicitly predicting the measurement outcome distribution and then sampling "state sample tomography" (SST; illustrated in Fig. 1).

This idea of "sample tomography" is similar to the "indirect" or "shadow" tomography discussed by Aaronson [24,25] that focuses on learning the measurement outcomes of a state rather than its explicit representation. He showed that if a "hypothesis state" can be found that is consistent with a set of observed two-outcome positive operator-valued measurements (POVMs) chosen from a (potentially unknown) distribution then almost all other measurement results from this distribution can be accurately predicted with the size of this observed set, scaling only linearly in the number of qubits [24]. However, finding this hypothesis state is still computationally problematic [24] and so we take things a step further; doing away with the hypothesis state and predicting the measurement results directly through other means (using a NN). Given that on current devices we can perform 2^n-outcome measurements in local product bases, we focus on building a generative model for n-qubit measurement distributions rather than predicting two-outcome POVMs. The overall philosophy is the same; if we can *implicitly* predict measurement outcome distributions (such that this process remains tractable for large systems) in arbitrary local bases and efficiently draw samples from these distributions, then we have achieved a useful form of tomography.

Many quantum states can be verified without full QST. For example, a known target state ρ_T can be decomposed onto a local operator basis (e.g., the Pauli basis $\rho_T = \sum_{\{\sigma_i\}} a_i\sigma_i$) and a form of the fidelity between the target state and the state produced on-device can be obtained by performing local measurements of these operators [$\mathcal{F} = \mathrm{Tr}(\rho\rho_T) = \sum_{\{\sigma_i\}} a_i\langle\sigma_i\rangle$]. Depending on the target state, this may require measurements in far fewer bases than are needed for full QST. The performance of quantum algorithms is also often quantified without full QST—the huge measurement and classical optimization cost of which makes it impractical for benchmarking on multiple noisy qubits. A highly relevant class of quantum algorithms to noisy intermediate-scale quantum (NISQ) computing are variational quantum algorithms, in which a parameterized ansatz circuit is optimized to prepare a state that minimizes an efficiently measurable (on a quantum computer) cost function. As these algorithms aim to eventually solve problems that require a classically intractable number of qubits, the prepared solution state of these algorithms is

rarely intended to be found exactly. Instead samples of the state in the computational basis (as in the quantum approximate optimization algorithm [26]), the minimum value of the cost function [as in the variational quantum eigensolver (VQE) [27]], or a circuit that produces the cost-minimizing state are the intended results of the algorithms. Having a circuit to prepare the solution state still allows it to be used in further simulations and algorithms or analyzed (even if full tomography is intractable).

Keeping in mind all these practical limitations, we build a generative model for local measurement distributions using what we call a basis-dependent RBM (BDRBM). This is a RBM neural network with parameters that are basis dependent, allowing the RBM's underlying probability distribution to continuously vary with the basis choice. A feedforward neural network (FFNN; sometimes also called a multilayer perceptron) is used to predict the RBM parameters for an input basis setting; these parameters implicitly determine the probability distribution represented by the RBM, and so in doing this we perform distribution prediction. Sampling prediction for the measurement basis is achieved through computationally efficient Gibbs sampling of the predicted distribution. By training the two networks using measurements of the quantum state in random local bases we can do sample tomography of the state efficiently. The relationship between the two networks is illustrated in terms of the BDRBM training process in Fig. 2. The FFNN can interpolate between measurement basis settings it has been trained on. This means that, as long as the function mapping measurement basis choice to the corresponding RBM parameters is sufficiently smooth, we can train on a small (compared to full tomography) number of random measurement settings to get a useful, sampling-based description of the state. Unlike density matrix MLE, due to the choices of network and training procedure, all stages of this process can be performed efficiently. The computational complexity of each training step for the RBM scales linearly with the number of qubits and the FFNN can be trained using efficient backpropagation. Having one neural network determine the parameters of another is an approach that is particularly well suited to quantum problems where we are trying to model complicated but smoothly varying probability distributions.

The BDRBM scheme broadly falls into the category of metalearning, where one attempts to solve a larger problem by looking at how different instances are learned [28]. In this case the FFNN learns how the RBM represents the measurement data in different local bases and tries to link them together. The approach exploits the efficiency in training and sampling from RBMs (as in Torlai's approach) while learning measurement distributions, not the state directly, to avoid needing optimization over complex parameters and reducing the required number of measurement settings. This then allows samples to be taken efficiently in arbitrary local bases (not just the computational), giving access to all the quantities that can be calculated from these while avoiding sums over exponentially many terms. Important such quantities are the expectation values of observables; these can be decomposed onto

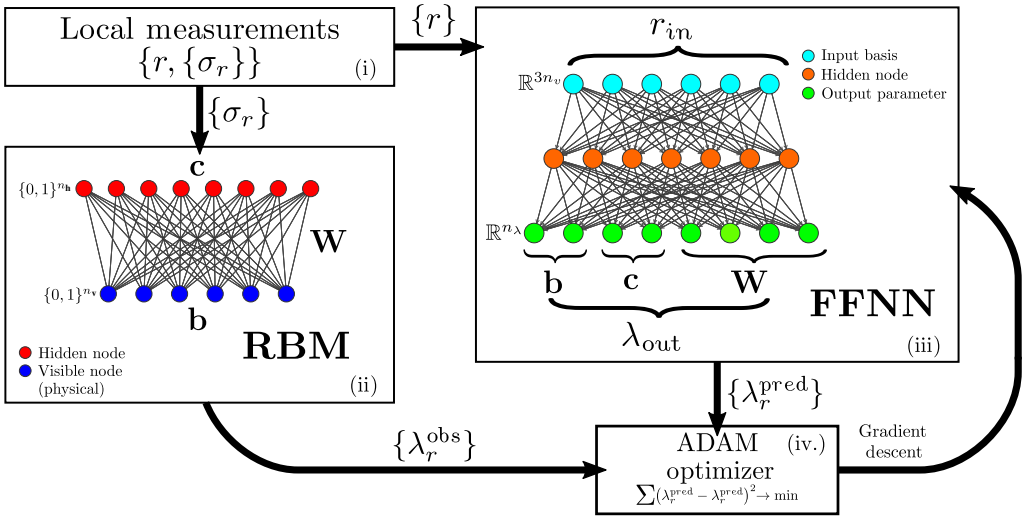

FIG. 2. Diagram of the BDRBM training process. (i) Measurements $\{\sigma_r\}$ are taken of the quantum system in local bases denoted by $\{r\}$, the Bloch sphere coordinates of the each qubits' measurement axis. The measurement bases are randomly chosen such that $\{r\}$ is uniformly distributed on the surface of the Bloch (upper hemi)sphere for each qubit. (ii) These measurement results are then used to train a RBM with the CD-1 algorithm; the RBM parameters are recorded as λ_r^{obs}. (iii) A FFNN (here shown with one hidden layer) takes the basis setting as an input and outputs a set of predicted RBM parameters λ_r^{pred}, thereby implicitly predicting the measurement distribution for this basis setting. (iv) The parameters in the FFNN are trained using gradient descent, minimizing the squared difference between λ_r^{pred} and λ_r^{obs}.

a set of local operators whose expectation values can be estimated efficiently through sampling in their eigenbasis. Cross-correlations of samples in random local bases [29] can be used to calculate other quantities, such as a form of quantum state fidelity as well as the purity of the state. As full tomography can be performed using local measurements taken in an informationally complete set of bases [2], by sampling from the BDRBM in this set of bases one could use it to perform full QST without having to make additional queries to the device (provided the BDRBM has learned enough about the state). As we only attempt to predict measurement outcomes, our method works in exactly the same manner for both pure and mixed states, the purity being reflected in the distributions used for training (and can be recovered with the sampling method in Ref. [29]).

II. RESULTS

A. BDRBM sample tomography

A BDRBM is a composition of two neural networks, a RBM and a FFNN. The FFNN converts measurement basis settings to a set of predicted parameters for the RBM. These parameters determine the probability distribution that the RBM succinctly represents—in this case an estimate of what the measurement distribution would look like in the local basis of interest. Once the RBM is passed the parameters predicted by the FFNN, the resultant distribution can be efficiently sampled from in an arbitrary local basis.

For a target state ρ (as viewed in the computational basis), we build our model to predict the state's measurement distributions in different local bases. The measurement outcomes for each qubit i are the ± 1 eigenstates $|0_b^{(i)}\rangle$ and $|1_b^{(i)}\rangle$ of a single-qubit operator $\hat{O}_b^{(i)}$, allowing the basis for that qubit to be defined in terms of the Bloch-sphere angles of the state $|0_b^{(i)}\rangle$, θ_i, ϕ_i. The overall basis is then specified by the set of these angles for each qubit $b = (\theta_0, \phi_0, \ldots, \theta_{n-1}, \phi_{n-1})$. We write the vector of n-qubit measurement outcome probabilities as $\mathbf{p}_b$; the jth component of this is the probability of the outcome $|j_b\rangle$, where j is expressed in binary [e.g., for three qubits, $(\mathbf{p}_b)_{110}$ is the probability of measuring $|1_b^{(2)}\rangle \otimes |1_b^{(1)}\rangle \otimes |0_b^{(0)}\rangle = |110_b\rangle$]. By rotating the state with a unitary $U(b)$ such that the basis b aligns with the computational axis, $\mathbf{p}_b$ is given by the diagonal of the resulting density matrix:

$$\mathbf{p}_b = \mathrm{diag}[U(b)\rho U(b)^\dagger], \tag{1}$$

$$U(b) = \bigotimes_{i=0}^{n-1} U_i(\theta_i, \phi_i), \tag{2}$$

$$U_i(\theta_i, \phi_i) = \begin{pmatrix} \cos(\theta_i/2) & e^{-i\phi_i}\sin(\theta_i/2) \\ \sin(\theta_i/2) & -e^{-i\phi_i}\cos(\theta_i/2) \end{pmatrix}. \tag{3}$$

The 2^n functions $\mathbf{p}_b$ are generally complicated as the unitaries $U(b)$ allow a potentially large number of off-diagonal terms to contribute. The analytic form of these equations is cumbersome and entirely dependent on the state in question, making them difficult to exploit in a general tomographic scheme, particularly when dealing with an unknown state. Instead, we note that they are continuous functions of $\{\theta_i, \phi_i\}$ and take advantage of the flexibility of neural networks as function approximators to try to infer them from measurement data. As we have moved away from the density matrix formalism, the approach comes with the caveat that while the BDRBM gives a valid probability distribution in any local basis, the distributions in different bases do not necessarily reconcile to form a valid quantum state. However, provided the BDRBM is sufficiently expressive and trained on enough data that it yields good accuracy in predicting the measurement outcome distributions for the state, then the state's properties should be (at least approximately) reproduced. The BDRBM approach assumes that the measurement distribution of the state in question changes relatively smoothly as the basis is varied, allowing interpolation between the settings used for training. We justify this by noting that the rotation matrices in Eq. (3) are smooth functions of the measurement basis angles $\{\theta_i, \phi_i\}$, and so measurements in a given basis provide information about other nearby bases. A basic requirement for this is that the training data cover each qubits' Bloch sphere sufficiently. The RBM can identify correlations in measurement data, but these correlations vary with the basis setting. This variation must be learned by the FFNN, and so enough training data are required that this can be done.

The measurement outcome distribution (for a given basis) in our BDRBM model is represented using a RBM. RBMs were introduced by Smolensky [30], and have become a staple tool in the ML community following Hinton's invention of the contrastive divergence training algorithm [19]. They consist of an undirected graph of two connected layers of binary-valued "neurons" or "nodes"; a "visible" layer of size $n_\mathbf{v}$ representing the states of the physical subsystems (in this case the qubit states after measurement in the basis of interest) and a stochastic "hidden" layer (of size $n_\mathbf{h}$). Each configuration of neuron states has an associated energy $E(\mathbf{v}, \mathbf{h})$ and a probability of occurring given by $p(\mathbf{v}, \mathbf{h}) = e^{-E(\mathbf{v},\mathbf{h})}$. Assuming that the distribution in question has some exploitable underlying structure, RBMs are a parameter-efficient way to encode a probability distribution compressing $2^{n_\mathbf{v}}$ outcome probabilities into $n_\mathbf{v} + n_\mathbf{h} + n_\mathbf{v} n_\mathbf{h}$ RBM parameters. The probability of a given configuration of visible neuron states $\mathbf{v}$ occurring is determined by the values of a set of learned RBM parameters λ; vectors $\mathbf{b}$ and $\mathbf{c}$ (bias terms on the visible and hidden neurons), and a weight matrix $\mathbf{W}$ that mediates

correlations in the distribution. The unnormalized probability $\tilde{p}(\mathbf{v})$ of a given configuration of visible neurons $\mathbf{v}$ is given by

$$\tilde{p}(\mathbf{v}) = e^{\sum_i b_i v_i} \prod_j (1 + e^{c_j + \sum_i v_i W_{ij}}). \tag{4}$$

The actual probability of this configuration occurring is then obtained by normalizing this quantity:

$$p(\mathbf{v}) = \frac{1}{Z}\tilde{p}(\mathbf{v}), \qquad Z = \sum_{\{\mathbf{v}\}} \tilde{p}(\mathbf{v}). \tag{5}$$

Direct calculation of this probability requires first calculating the partition function Z—as this involves a summation over $2^{n_\mathbf{v}}$ possible configurations of $\mathbf{v}$, this becomes intractable for large systems (large $n_\mathbf{v}$). A computationally efficient algorithm was proposed by Hinton [19] to optimize the RBM parameters $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{W}$ to minimize the Kullback-Leibler divergence between the model's probability distribution and the distribution of a set of training data. This is done by making use of a Markov chain Monte Carlo (MCMC) process that efficiently generates samples from the RBM's represented distribution while avoiding calculation of Z [14]. The time taken for these samples to converge to the RBM's underlying distribution (the asymptotic distribution of the MCMC process) varies depending on the implementation of the RBM, the distribution of the starting samples, and the complexity of the underlying distribution, and so we will not cover it in detail here; it is however an important topic of ongoing research [19,31]. Crucially, each step of the MCMC process scales only as $\mathcal{O}(n_\mathbf{v} n_\mathbf{h})$, and so the training and sampling processes can be extremely computationally efficient provided that $n_\mathbf{h}$ is not too large. RBMs are capable of approximating arbitrary probability distributions, provided that the target distributions are sufficiently smooth and regular [32]. However, this may require an exponentially large number of hidden neurons, making the representation impractical [32]. This places some practical limitations on what a BDRBM is capable of, but as most states of interest in many-body physics possess numerous symmetries or underlying structure, this regularity condition seems not too great a requirement.

We increase the representational power of the network by allowing the RBM's parameters to change with the basis choice, $\lambda = (\mathbf{b}, \mathbf{c}, \mathbf{W}) \to \lambda(\mathbf{r}) = [\mathbf{b}(\mathbf{r}), \mathbf{c}(\mathbf{r}), \mathbf{W}(\mathbf{r})]$. To do this in a generalized framework, we use a FFNN. This is a simple but highly expressive neural network architecture consisting of layers of real-valued neurons with (forward-flowing) connections between them. The values of neurons in each layer are calculated by acting with an (usually nonlinear) activation function on the weighted sum of the neuron states in the previous layer, allowing an input state to be converted to an output state. In this case the FFNN takes as its input desired measurement basis and outputs a set of RBM parameters. We smoothly parameterize the basis on each qubit using the Bloch sphere coordinates of the normalized measurement axis vector, $\mathbf{r}_i = (x_i, y_i, z_i)$; a more concise description would instead use the polar angles of this axis; however, Cartesian coordinates are found to work more effectively (while also avoiding discontinuities). The output of an m-layer FFNN with $(n_{l_1}, \ldots, n_{l_m})$ hidden neurons in each layer, $\lambda(\mathbf{r})$, is then given by

$$\lambda(\mathbf{r})_i = \lambda_i^{\text{out}} + K_{ij}^{\text{out}}[(g_m \circ g_{m-1} \circ \cdots \circ g_1)(\mathbf{r})]_j, \tag{6}$$

$$g_k : \mathbb{R}^{n_{l_{k-1}}} \to \mathbb{R}^{n_{l_k}}; \mathbf{s} \mapsto f_k(\mathbf{a}_k + \mathbf{K}_k \cdot \mathbf{s}), \tag{7}$$

i.e., a composition ("$\circ$" is the composition operator) of functions $\{g_k\}$ that map the input of one layer to the output of the next. Overall, this maps the input $\mathbf{r}$ to an output $\lambda(\mathbf{r})$. Each g_k involves an elementwise activation function f_k, a bias vector $\mathbf{a}_k$, and a weight matrix $\mathbf{K}_k$. To allow the FFNN to act as a universal function approximator, these activation functions should in general be nonlinear. Experiments reveal that, while a linear FFNN is effective in many cases, the popular *leaky_relu* function, $g(x) = \max(x, 0.2x)$ [33], also performed well (allowing for some degree of nonlinearity).

The exact form of the FFNN for a given state, i.e., number and size of the hidden layers and their activation functions, should be chosen on a case-by-case basis. As previously discussed, while the networks produce valid probability distributions in arbitrary local bases, these will be an approximation to the statistics of the density matrix and may not exactly correspond to a positive-definite quantum state. On the other hand, the distributions in bases that are different from the computational basis are more directly accessible than when using a density matrix as the RBM parameters that define this can be obtained with a single pass through the FFNN, rather than a computationally expensive rotation of the state. This means that quantities that can be calculated using samples in local bases (e.g., state entropy, certain state fidelities, or physical observables) may be estimated more tractably with a BDRBM than would be feasible with exact calculations on a large density matrix.

Training the network takes place in several stages, as illustrated in Fig. 2. First, measurements are performed on the quantum system in a number of local bases; to ensure that the FFNN can predict across the desired space of basis settings, these bases must reasonably well cover the Bloch spheres of each qubit. We observe that the network predicted distributions most accurately when the bases are selected at random uniformly across each qubit's Bloch

sphere. Points on opposite sides of the spheres give equivalent measurements (up to relabeling), meaning that the measurements can be restricted to only the Bloch upper hemispheres, reducing the amount of data needed to learn $\lambda(\mathbf{r})$. The basis settings are retained to form the inputs for the FFNN's training data. For each measurement, the RBM is trained to find a compressed representation of the measurement data (using Hinton's CD-1 algorithm [19]) and the observed RBM weights and biases for each basis are recorded to form the outputs of the FFNN's training data $\{\lambda_r^{\text{obs}}\}$. To ensure similarity and smoothness between observed RBM parameters from basis to basis, and to speed up convergence, the parameters found for one basis are used as the starting point for the next. We use $L2$ regularization (adding a cost penalty to the summed square weight elements) during this stage to prevent the RBM weights growing too large; when done correctly, this both speeds up and improves convergence [34] while leading to more manageable variation within the FFNN's training data.

Finally, regression is performed using gradient descent (the ADAM algorithm was used in the examples given [35]) on the observed RBM parameters to fit the FFNN, minimizing the ℓ^2 norm between observed and predicted parameters (along with some $L1$ regularization of the FFNN weights to reduce overfitting and allow clearer analysis of the weights). An iterative fine-tuning process is found to reduce overfitting and lead to better accuracy in the predicted distributions; this is done by retraining the RBM on the measurement data using the FFNN's predictions for each training basis as the RBM's starting point. RBMs are very flexible models and so can find many different but similarly faithful representations of a distribution. It is likely that during the initial RBM optimization step parameters will be observed that may not reconcile well with the rest of the data set. Iteratively predicting and relearning the RBM parameters allows the observed parameters to drift closer to a set that can be most effectively described by the FFNN.

It is often the relative sizes of the RBM parameters that are most important in representing a given distribution, and so the parameters often exhibit a large amount of covariance. This can be exploited in an optional preprocessing step prior to the regression using principal component analysis (PCA). PCA finds the linear combinations of RBM parameters that contribute the most to the covariance of the data and ranks them in terms of this contribution [36]. By rewriting the RBM parameters in terms of their projections onto these directions, the regression process can converge more quickly and accurately. In addition, by only retaining the projections onto the directions that explain the largest amounts of covariance, one can greatly reduce the dimension of the training data, potentially without losing much useful information [36]. This results in a smaller FFNN being required, making both the regression step easier and leading to a more concise final representation of the state.

B. Classical fidelity for noisy quantum device transverse-field Ising model generation

To demonstrate this tomographic process on a real quantum system, we apply it to random local basis measurements taken for a set of six-qubit states prepared on the *ibmq_singapore* and *ibmq_paris* devices [37] (run in early April 2020). These states are prepared using circuits that are optimized with a tensor-network-based method (using the quimb PYTHON package [38]) to output a state that, in the absence of any device errors, matches the ground state of the one-dimensional (1D) antiferromagnetic transverse-field Ising model (TFIM), to within a quantum infidelity of order 10^{-5}. This is done by expressing an ansatz circuit in terms of a tensor network and then taking advantage of tensor contraction schemes to maximize the fidelity between the target ground state and the output of the circuit. The antiferromagnetic TFIM Hamiltonian used here for a 1D spin chain with spins on the ith sites S_i^x and S_i^z along the x and z axes, respectively, is given by

$$H = J_z \sum_{\langle i,j \rangle} S_i^z S_j^z - J_x \sum_i S_i^x, \tag{8}$$

where the first sum is performed over nearest-neighbor pairs $\langle i,j \rangle$. The system undergoes a phase transition when J_z (the interaction strength) and J_x (the transverse field) are equal in magnitude. This model provides a good test case for the tomographic process as by varying the relative sizes of J_z and J_x we can examine its performance as the ground state goes from the Bell-like state ($|\psi_0\rangle \sim |01\rangle^{\otimes n/2} + |10\rangle^{\otimes n/2}$) for small J_x to the nonentangled product state ($|\psi_0\rangle \sim |+\rangle^{\otimes n}$) via an intermediate state that exhibits long-range entanglement. The BDRBM utilized here consists of a RBM with six hidden neurons and a simple linear FFNN (consisting only of input and output layers) that outputs RBM parameters, λ^{out}, as

$$\lambda_i^{\text{out}}(\mathbf{r}) = \lambda_i^0 + \sum_j M_{ij} r_j . \tag{9}$$

Training the BDRBM to recreate measurement outcome distributions for data in bases that it has already seen is not enough for useful tomography. It is also necessary that the predicted distributions for unseen bases are accurate and so we need to be able to test how well this is done. If we were using a density matrix ansatz then we could simply calculate the fidelity between the learned and target states; however, as discussed in Sec. I, the size of the density matrices makes this impractical. To check that the network's performance on unseen data is similar to that on the training data, we use a simple cross-validation

method. This involves retaining a portion of the measurement data (i.e., not using the data for these bases to train the BDRBM) and comparing the observed probabilities of each outcome to the predictions for the held-out measurement bases. There are several possible figures of merit for the comparison between the predicted and observed distributions. The one that is used here is the classical version of the quantum fidelity of a pure state (the quantum fidelity can be shown to be the maximization of this quantity across all possible POVMs). The "classical fidelity," $\mathcal{F}_c$ (called the Bhattacharyya coefficient in classical statistics), expresses the similarity between two probability vectors $\mathbf{p}$ and $\mathbf{q}$, and is given by

$$\mathcal{F}_c(\mathbf{p},\mathbf{q}) = \sum_i \sqrt{p_i q_i}. \tag{10}$$

The quantum fidelity can be estimated with local measurement samples taken from the BDRBM using the method described in Ref. [29] and so could be used for cross-validation. However, for this purpose, we use the classical fidelity as it is significantly faster to calculate. It also allows us to make comparisons between the BDRBM's predictions and the measured device data on a basis-by-basis level. This makes it well suited to the cross-validation used to identify if overfitting has occurred as it gives us two well-defined standalone figures of merit for the training and validation sets. The quantum fidelity is a combined comparison for all bases and so, as we are also dealing with a generative model for measurements rather than a density matrix, is not necessarily any better suited for this task. As shown in Ref. [17], while the classical fidelity provides only an upper bound on the quantum fidelity, it provides an effective substitute; a high classical fidelity usually indicates a high quantum fidelity.

Circuits are run on the *ibmq_singapore* and *imbq_paris* devices to prepare the ground states of the six-qubit antiferromagnetic TFIM with varying J_x (and constant $J_z = 1$), as in Eq. (8). These states are then measured in 200 random local bases (uniformly distributed on qubit upper Bloch hemispheres). For each device, the results are split into a training set and a validation set, and the training data are fed into the tomographic algorithm as detailed in Sec. II A. As we are considering only a relatively small system, we compare the exact predicted probability distributions [using Eq. (5)] to the empirical distributions in the measured data. For larger systems, these comparisons could instead be made tractably (although with some introduced sampling error) between samples predicted by the BDRBM and the measurement data. To compare the performance of the algorithm for different transverse-field strengths, the average classical fidelities across the bases in the training and validation sets are calculated between the observed probability distributions in the measurement data and the BDRBM's distribution predictions. As these are NISQ devices, it is also useful to see how well the BDRBM's predictions match what would be expected in the absence of device noise, and so average classical fidelities are also found between the ideal measurement distributions for the target state and the BDRBM's predicted distributions for the bases in the two data sets. Overfitting by the FFNN can be quantified by looking at the extent to which the reconstructive fidelity (for the training data set) is higher than the predictive fidelity (for the unseen validation set); this is a basic form of cross-validation.

The achieved average classical fidelities $\overline{\mathcal{F}_c(\mathbf{p}_{\text{obs}}, \mathbf{q}_{\text{RBM}})}$ (the mean of classical fidelities for the bases used in the comparison) for simulated (ideal) data and for the data collected from *ibmq_singapore* and *imbq_paris* are shown in Fig. 3. These are calculated between the observed measurement distributions $\mathbf{p}_{\text{obs}}$ and the BDRBM's predicted distributions $\mathbf{p}_{\text{RBM}}$ for the bases in question as in Eq. (5).

Figure 3(a) shows that this simple linear BDRBM performs well for an ideal, simulated preparation of the state. Both the reconstructive and predictive fidelities are high with a small degree of overfitting indicated by the reconstructive fidelity being slightly higher than the predictive. We see a dip in both fidelities at a transverse-field strength of around $J_x = J_z = 1$; at this point the ground state of the antiferromagnetic TFIM undergoes a quantum phase transition from an ordered to disordered phase (as viewed in the computational basis) via a state with long-range entanglement. A dip in fidelity around this phase transition is expected as the more complicated entanglement structure means that in the state's Schmidt basis (the product basis with the minimum number of terms in its description of the state) a larger number of significant terms are present. All of these terms contribute to the function that maps basis choice to outcome probabilities [Eq. (1)], making this function more complicated and therefore harder to learn.

For both NISQ devices [Figs. 3(a) and 3(b)], we see that the predictive fidelity is consistently high between the BDRBM's predictions and the withheld measurement data (labeled val/measurement in Fig. 3) at around 97%–98%. This is true even at transverse fields around the phase transition where the BDRBM trained with ideal simulated data only manages to achieve predictive fidelities of around 95%. This indicates that the true states on the devices at these points are easier to learn than the intended highly entangled ground state, showing that the devices may not be able to reliably produce the long-range entanglement required.

When fed data taken on *ibmq_singapore*, the model reproduces both the training (reconstruction) and validation (prediction) sets' observed measurement distributions with a high classical fidelity [Fig. 3(b)]. However, these BDRBM-predicted measurement distributions do not agree particularly well with the measurement distributions of the ideal target state. The fidelities between these distributions are a lot lower and this does not improve as the

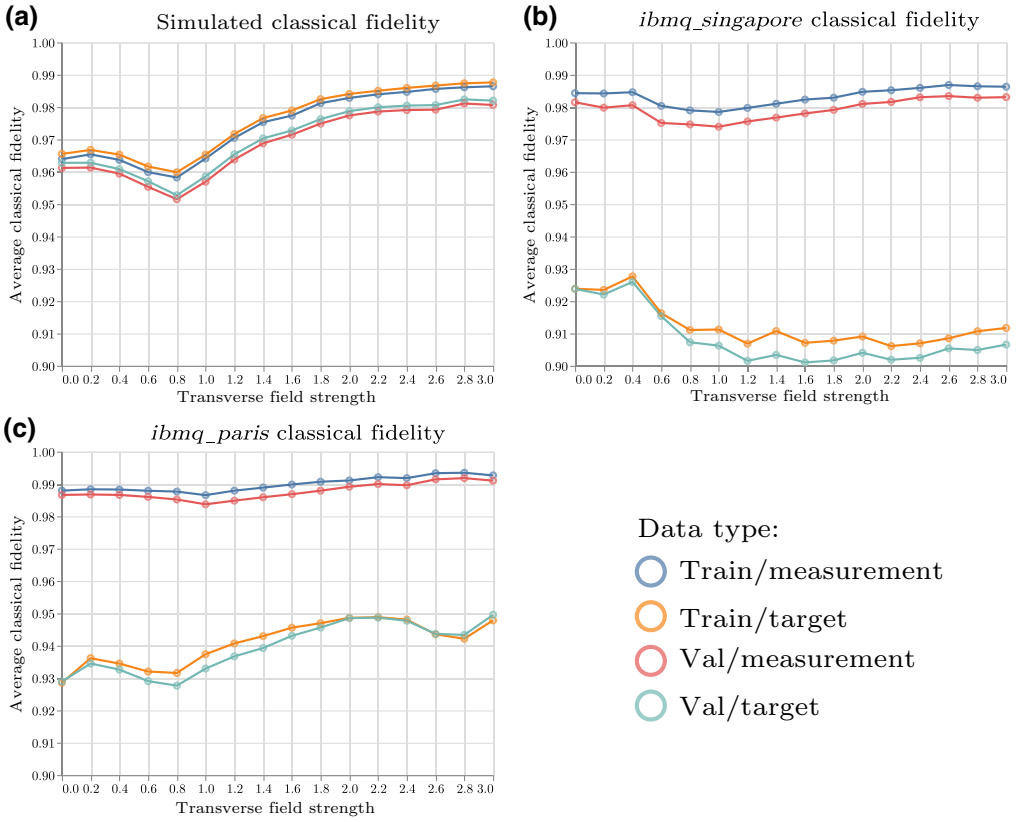

FIG. 3. Achieved classical fidelities for BDRBM tomography of six-site antiferromagnetic TFIM. These graphs show the average classical fidelities between the predicted measurement distributions of the BDRBM after training with (a) ideal simulated data, (b) data from the *ibmq_singapore* device, and (c) data from *ibmq_paris* over a range of transverse fields (and $J_z = 1$). The training process is as described in Sec. II A, with six hidden neurons in the RBM and a linear FFNN with directly connected input and output layers. Each graph shows four fidelities between the BDRBM's predicted distributions and train/measurement, the observed measurements in the bases used for training; train/target, the exact simulated target distributions for the training bases; val/measurement, the withheld measurements in the bases used for validation; val/target, the exact simulated target distributions for the validation bases.

transverse field gets large (despite the target ground state becoming much simpler). This discrepancy indicates that the tomographic algorithm has managed to accurately learn the state of the system, incorporating the noise that leads to an imperfect preparation of the desired state. As the ground states for larger transverse fields (past the phase transition) get closer to the $|+\rangle^{\otimes n}$ state, this state consists of a superposition of all possible physical states on the device (the computational basis). In the presence of a large amount of dephasing noise, for example, due to crosstalk between qubits or large numbers of noisy two-qubit gates, these states could prove fragile. If this noise is correlated (for example, as a result of a state-dependent crosstalk changing the locally experienced fields of neighboring qubits) then the more highly entangled states (ones with low J_x) may be more resistant to it, leading to the higher target-prediction fidelity seen in the data for low J_x. The circuits used to generate these states consist of an ansatz with a depth (the number of controlled-Z gates acting between each adjacent qubit pair) of 3; this is overly complex for the relatively simple, almost separable ground states at large transverse fields and so excessive noise due to errors on the two-qubit gates may have either ruined the superposition by the time the measurements take place or entangled the state more than is necessary.

The *ibmq_paris* device appears to fare much better [Fig. 3(c)] than *ibmq_singapore*. A discrepancy is still observed between the experimental data-prediction and target-prediction fidelities, implying that a good deal of noise is still present; however, this discrepancy is smaller than for *ibmq_singapore*. Crucially, the performance of the tomography increases as the transverse field becomes large, qualitatively agreeing with the trend observed in the simulated data. This indicates that the newer device is considerably better at maintaining coherences between its computational basis states. A reduction in fidelity is observed for the largest values of J_x; as mentioned before, this is likely due to the circuit ansatz being used containing more gates than is strictly necessary for the simple states in question. As the same depth-3 ansatz is used for these circuits, we again have an excessive number of two-qubit gates that will not aid greatly in the state preparation and,

as these gates have much higher errors than single-qubit gates, these are likely the primary cause of the suppression of the fidelity for large J_x.

C. Scaling

Analyzing how many measurement settings are required for this tomographic process is difficult as there are many factors that will increase the complexity of the learning process. Broadly speaking, it will depend on the complexity of the state in question; it is expected that the number of measurements required will increase with the number of significantly contributing terms in the state's local Schmidt basis, as this makes Eq. (1) more complicated. The suitability of an RBM in representing the target state in local bases is also important. While RBMs are capable of closely approximating arbitrary probability distributions, this may require an infeasibly large number of hidden neurons, making the learning and sampling processes too slow to be practical. The RBM must be able to well approximate the state in an arbitrary basis, but it must also do so with relatively smooth changes of the parameters (which can be learned by the FFNN); this may also increase the number of hidden neurons required. This issue of parameter smoothness could be partially alleviated by using a more complicated FFNN; however, this may then increase the number of parameters to be trained and so require more training data. If dimensionality reduction is to be performed using PCA then the number of sets of training RBM parameters (and so the number of measurement settings) must be greater than or equal to the number of parameters in the RBM ($n_{\mathbf{v}} + n_{\mathbf{h}} + n_{\mathbf{v}} n_{\mathbf{h}}$), and so this puts a lower bound on the number of measurements required if this is to be used.

To investigate the scaling of the required number of measurement settings with the size of the state in question, a linear BDRBM (as described in Sec. II B) is trained to learn the antiferromagnetic TFIM ground state. The size of the state and the number of simulated measurements being passed to it are varied. The transverse field used is $J_x = 1$, corresponding to the critical point of the phase transition and the region that is hardest for the network to learn. To allow the model complexity to grow suitably with the increasing size of the state, the number of hidden neurons is set to be equal to the number of sites (this gives the best predictive fidelity). The achieved predictive and reconstructive fidelities for different system and training data sizes are shown in Figs. 4(a) and 4(b). Figure 4(c) gives the difference between the reconstructive and predictive fidelities. This indicates the amount of overfitting that has occurred.

The fidelity is seen to first improve as the training set gets larger but then briefly decreases before a further rapid increase leads to saturation. Error curves of this shape (or a fidelity curve in this case) are well documented in deep learning (and in overparameterized regression more generally) as a phenomenon known as "double descent" [39,40]. As more training data are added, the model's performance improves up to the point where the overparameterized neural network starts to fit the training data too closely and the generalization error increases [these are the drops in predictive fidelity in Fig. 4(a)]. Eventually the "interpolation threshold" is reached and the model uses its excess parameters to interpolate directly between data points, giving the maximum amount of overfitting [the peaks in Fig. 4(b)] [40]. However, past this point the inductive bias due to the regularization procedure used (in our case $L1$ regularization) comes into play and any added data allow the stochastic gradient descent algorithm to find the predictors that give as simple a model as possible without sacrificing too much reconstructive performance. This "Occam's razor"-like approach then results in a greatly improved generalization error [the rapid improvement in predictive fidelity seen in Fig. 4(a) following the dips] [39].

Using this linear model, the predictive fidelity saturates at lower values as the number of sites grows; this implies that a more expressive model is needed to more accurately represent large ground states of this type. However, the achieved average fidelity of around 93% means that, given the size of the system, the predicted measurement distributions will exhibit the significant features of the true probability distribution, with peaks and dips in the distributions located in the correct places. This reasonably faithful approximation of the state can be achieved with only of the order of 100 measurement bases and, in terms of the measurements taken, makes no assumptions about which terms in the density matrix (and so which measurement bases) might be relevant. In a NISQ setting where noise processes lead to substantially mixed states, full characterization would require full tomography to be performed and this would instead require $3^{10} = 59\,049$ measurements.

D. TFIM filters for linear BDRBM

When using neural networks, it is often illuminating to look at the filters (the weight matrix $\mathbf{M}$ in this case) that the model has inferred from the data to try to draw some conclusions about the underlying system. For the simple linear model used for the TFIM data, this weight matrix is easy to interpret; the element M_{ij} gives the derivative of the RBM parameter i with respect to the basis coordinate j, $M_{ij} = \partial\lambda_i/\partial r_j$, and so tells us how strongly the parameter depends on that coordinate. In this section we analyze the filters found from the tomography in Sec. II B and discuss how these relate to the state that is learned.

While the nonlinearity of a RBM makes it difficult to directly interpret its parameters, we can still make some qualitative statements. Broadly speaking, the visible biases, $\mathbf{b}$, control the probability of their associated neurons taking the state 1 (or $|1\rangle$ if these neurons represent

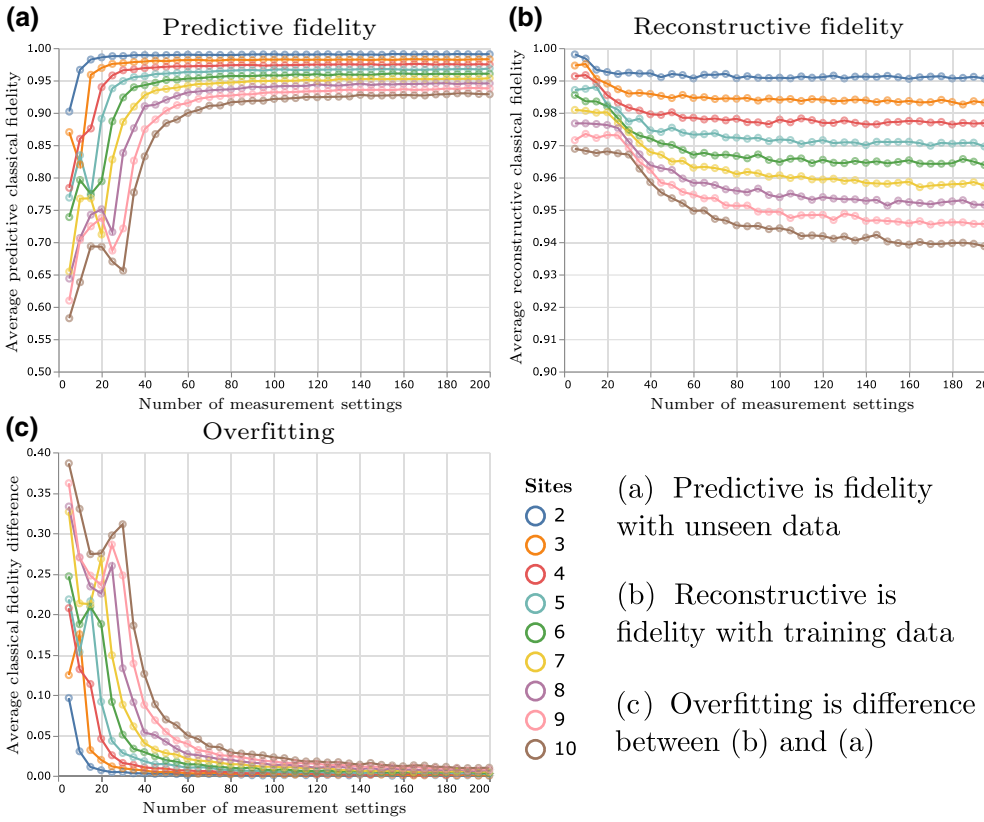

FIG. 4. Scaling of fidelities and overfitting with the number of training measurements. These graphs show how performance of our scheme changes as the BDRBM is given larger sets of training data. This is shown for a linear BDRBM trained with different numbers of simulated measurements (noiseless with 8192 shots per measurement basis) of the antiferromagnetic TFIM ground state with $J_x = J_z = 1$ (the quantum-critical point). These simulations are performed for ground states with between two to ten sites (indicated by the line color). The RBMs used have $n_{\mathbf{v}} = n_{\mathbf{h}}$. (a) The scaling of the average predictive classical fidelity between the BDRBM's predicted measurement distribution and the exact (calculated from the target state's state-vector) distributions in randomly selected bases that are not used for training. (b) The scaling of the average reconstructive classical fidelity between the BDRBM's predicted distribution and the exact distributions in the bases used for training. (c) The scaling of the difference between the average reconstructive and predictive classical fidelities—a measure of how much the network has overfitted to the training data.

a qubit state) [34]. For a product state, the reduced state of a given qubit is independent of state of the others, and so these could therefore be represented using a RBM with no hidden neurons and visible biases that vary only with the measurement basis of their own associated qubit. The weights and hidden biases have the role of mediating correlations between outcomes and so are needed to express more complicated states. Nonlocal correlations may also be indicated by visible biases (a parameter that typically defines local properties) that are dependent on the measurement settings of qubits other than the one they are associated with.

Following these simple qualitative interpretations, one would expect that, for the antiferromagnetic TFIM ground states with $J_x < J_z$, the measurement distributions would be most strongly dependent on the z coordinates of each qubit's basis and so the nonzero components of $\mathbf{M}$ will mostly lie along rows corresponding to z coordinates. This is because the z axis corresponds to a Schmidt basis for the state in this ground state. Measurements taken along axes close to this result in sparse distributions with sharp peaks; distributions with many zeros and a few large peaks require the largest weights and biases for the RBM and so these axes (and therefore components along them) end up being the most important in determining the RBM parameters. In this regime the state is entangled and so we also expect to see that the columns of $\mathbf{M}$ that correspond to hidden weights and biases contain nonzero values as these RBM parameters will be required to express correlations in the measurement outcomes (and must be able to vary with the local basis choices). The filters that we expect for large transverse fields are much simpler; as there is little entanglement in these cases, we expect that the dominant terms in $\mathbf{M}$ will correspond to visible biases and that these should depend only on the basis choice for their associated qubit. As these ground states are close to x axis eigenstates

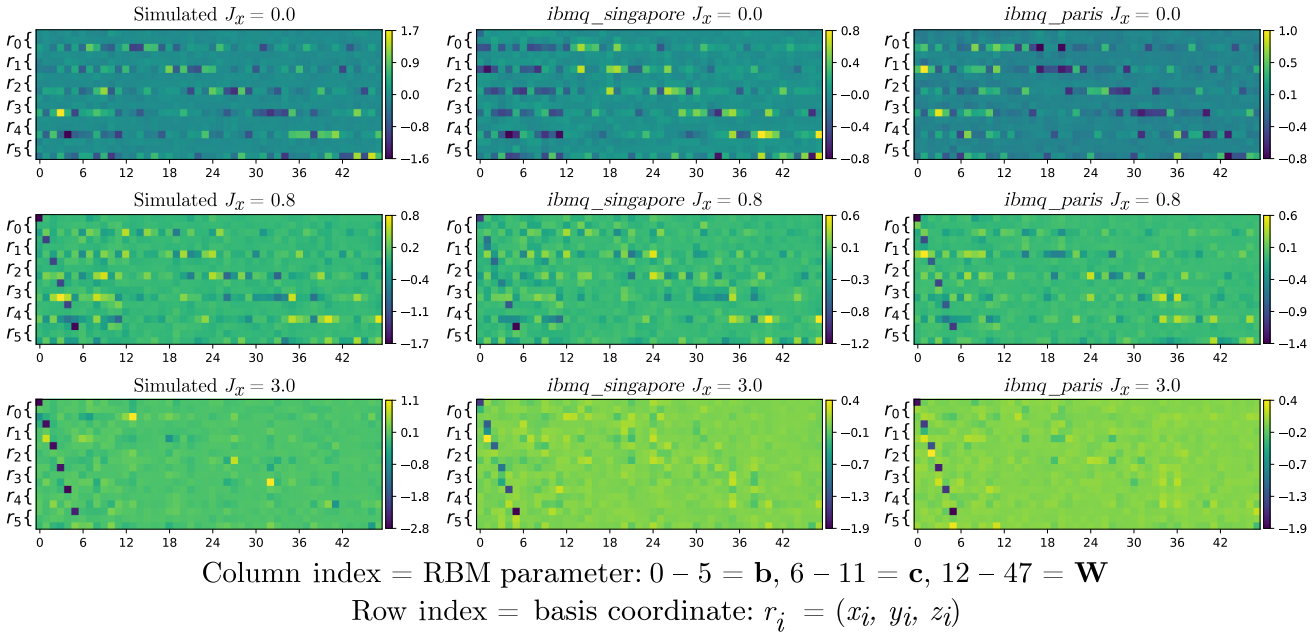

FIG. 5. Filters found for linear FFNNs trained on antiferromagnetic TFIM data. These are visual representations of the component values of the FFNN filters found for some of the TFIM ground states in Sec. II B. As a linear network is used, each component of a filter M_{ij} gives the derivative of the predicted RBM parameter indexed j (columns) with respect to a Bloch sphere coordinate i (rows). Here the rows are grouped into coordinates $r_k = (x_k, y_k, z_k)$ to indicate which basis coordinates are associated with each qubit. A range of transverse fields are shown: $J_x = 0$ being in the ordered, entangled phase; $J_x = 3$ in the disordered low-entanglement phase; and $J_x = 0.8$, which is in the intermediate phase around the phase transition. The RBM parameters (columns in each image) are indexed as follows: 0–5 are the biases on each visible neuron (site), 6–11 are the biases on each hidden neuron, and 12–47 are the (flattened) weights between visible and hidden neurons.

$(|+\rangle^{\otimes n})$, we would expect the x values of the basis choice to become significant, in particular determining the visible biases of their respective qubit.

The filters that are found for the simulated, *ibmq_singapore*, and *ibmq_paris* data are shown in Fig. 5 for three values of J_x (corresponding to the ordered, intermediate, and disordered phases). What is observed for the simulated data is broadly in line with predictions. For low transverse fields, the majority of the filters' nonzero components link the RBM parameters to the z coordinates of the qubits' measurement bases. In this regime significant weights between basis coordinates and hidden parameters (weights and biases) are present and the visible biases (the first six columns of each filter) appear to have a delocalized dependence on the basis choice; these are all qualitative indicators of correlations, with the latter delocalization perhaps being an indicator of entanglement. As J_x increases, we see an increasingly prominent local dependency of the visible biases on the x coordinates of each qubit's measurement basis; in the intermediate regime both the nonlocal z dependencies and local x dependencies are present, while for large transverse fields, the filter components for the hidden weights and biases become suppressed and the dominant terms lie solely on the visible biases.

Perhaps more interesting are the filters that are found for the two NISQ devices. The *ibmq_singapore* device performs worse for large transverse fields than small ones. This is reflected in the filters as, while the filter found for $J_x = 0$ matches qualitatively well to that of the simulated data (with the same matrix elements being nonzero, although not necessarily having the exact same values), the filters for the larger fields begin to diverge quite dramatically. At large fields we still observe sizeable basis dependence of the hidden weights and biases (whereas these parameters should be close to zero), indicating that extra undesired correlations are present on the device. The visible biases in this region have a less clear basis dependence than for the simulated data; rather than strongly depending on $\{x_i\}$ and $\{z_i\}$ (rows $0, 3, 6, \ldots$ and $2, 5, 8, \ldots$, respectively, for each filter) the y coordinates (rows $1, 4, 7, \ldots$) of the bases also appreciably contribute. Unfortunately, as we are only looking at one family of states and the RBM parameters are hard to directly interpret, it is difficult to conclusively pin down which processes are responsible for these discrepancies. The unexpected components corresponding to RBM weights (and so measurement correlations) could imply some kind of correlated noise process (for example, crosstalk between qubits), whereas the smeared out visible bias basis dependence could indicate

a locally acting error imposed by a qubits' environment (a systematic single-qubit rotation).

As seen in Sec. II B, the *ibmq_paris* device is a lot better at generating the desired ground states. Little qualitative difference is seen between its filters and *ibmq_singapore*'s for low transverse fields (both agreeing reasonably well to that of the simulated data). Encouragingly, as the transverse field grows larger, the *ibmq_paris* device manages to do a decent job of correctly introducing the local x dependencies of the visible biases and then damping down the correlations between qubits by reducing the basis dependence of the weights and hidden biases. As shown in Fig. 3(c), *ibmq_paris* does not perfectly generate the desired states and this manifests itself as small differences in the filters. While it appears that the correct qualitative dependencies are present, the values of the components do differ and these differences are magnified by the nonlinearities in the underlying RBM, leading to the predictive fidelity being around 5% higher for the measured data than for the ideal case.

III. DISCUSSION

In this work we have discussed a method by which machine-learning techniques may be used to perform quantum state sample tomography, a quantum state characterization scheme that we posit is a far more tractable alternative to full QST. Characterization of quantum states is an extremely useful tool in the analysis of NISQ devices; however, full tomography is prohibitively expensive in terms of the number of measurements required. The density matrix that full QST produces is also an unwieldy object for large systems due to its exponentially many parameters. Instead, we relax the requirement that a density matrix is produced and build a model for how the measurement distributions of the system vary with the choice of measurement basis, doing so in such a way that these distributions can be sampled from efficiently. The model is trained with measurements in random bases, making as few assumptions about which measurements must be taken as possible. This allows it to perform with good predictive accuracy on NISQ devices where unpredictable errors and imperfect qubit control may move the state into unexpected regions of the Hilbert space. As no assumptions are made about the purity of the state (although this can be estimated through sampling [29]), it is not restricted to pure states, furthering its applicability to NISQ devices.

Two types of neural network architecture are combined to yield the BDRBM, a trainable model for the measurement distributions. Provided the state possesses enough underlying structure, the model used to learn and represent the measurement distributions, a restricted Boltzmann machine, allows distributions for even very large systems to be expressed succinctly and sampled from efficiently. To account for the basis dependence of measurements that is integral to quantum mechanics, the RBM model is controlled by a second trainable neural network (a FFNN) that assigns parameters to the RBM that depend on the input basis setting. "Metalearning" is an area of active interest, particularly in ML applications where only small datasets are feasibly available [28]. This idea that one model's parameters can be training data for another model broadly fits into this field; however, it appears particularly well suited to the continuously varying measurement outcome distributions present in quantum mechanical systems.

We have also demonstrated how analysis of the learned parameters for the FFNN can be used to infer qualitative properties of the state. Comparisons between parameters learned for a simulated ideal state preparation and for those learned from a noisy device could be a useful tool in performing error analysis without having to perform prohibitively expensive process tomography. It also raises the question of whether it is possible to identify the Schmidt basis for a given state by iteratively relabeling the computational axis until the FFNN weights of the BDRBM representation of the state are maximally aligned along this direction, provided the state can be expressed using a sufficiently simple FFNN. By interpolating between the measurements used for training, this tomographic scheme attempts to get as much information out of each measurement as possible, allowing fewer measurements to be taken while still arriving at a useful generative model of the state's measurement statistics. This can then be either analyzed directly, used to predict distributions in unseen bases, or efficiently draw samples in these bases. The predicted samples could then be used as an intermediate stage in optimization schemes, allowing relevant quantities to be calculated without having to take additional measurements.

The scheme we have presented here has the potential to complement existing quantum algorithms. For example, the ability to sample from the prepared state in arbitrary local bases would be of great use in the VQE algorithm [27], allowing estimates of the energy to be made while also providing extra information about the prepared state. Using the technique given in Ref. [29], one can use samples in random local bases to estimate quantities of the form $\mathrm{Tr}(\rho_1\rho_2)$, giving access to state fidelities of this form, the purity of the state, and the entanglement properties of the state (through Renyi entanglement entropies). Crucially, the ability to efficiently estimate the overlap between prepared states as well as their energy expectation values with only a modestly sized set of random measurements could allow excited state generalizations of VQE [41] (based on projector methods) to be performed more efficiently than is currently feasible on-device—such algorithms are highly relevant to quantum chemistry and materials simulations. This means that not only does our scheme provide a pathway to more efficient benchmarking

and analysis of large device-prepared states, but can also complement existing near-term algorithms.

ACKNOWLEDGMENTS

This work is supported by the UK EPSRC (EP/P510257/1), the EPSRC Hub in Quantum Computing and Simulation (EP/T001062/1), the Royal Society, and the Samsung GRP grant. We acknowledge the use of IBM Quantum Services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team.

[1] J. G. Titchener, M. Gräfe, R. Heilmann, A. S. Solntsev, A. Szameit, and A. A. Sukhorukov, Scalable on-chip quantum state tomography, npj Quantum Inf. **4**, 19 (2018).

[2] J. B. Altepeter, D. F. James, and P. G. Kwiat, in *Quantum State Estimation. Lecture Notes in Physics*, Edited by M. Paris, and J. Řeháček (Springer, Berlin, Heidelberg, 2005), vol. 649, p. 105.

[3] D. M. Appleby, Symmetric informationally complete measurements of arbitrary rank, Opt. Spectrosc. **103**, 416 (2007).

[4] M. Wiesniak, T. Paterek, and A. Zeilinger, Entanglement in mutually unbiased bases, New J. Phys. **13**, 053047 (2011).

[5] T.-C. Yen, V. Verteletskyi, and A. F. Izmaylov, Measuring all compatible operators in one series of single-qubit measurements using unitary transformations, J. Chem. Theory Comput. **16**, 2400 (2020), pMID: 32150412.

[6] D. F. V. James, P. G. Kwiat, W. J. Munro, and A. G. White, Measurement of qubits, Phys. Rev. A **64**, 052312 (2001).

[7] Z. Hradil, Quantum-state estimation, Phys. Rev. A **55**, R1561 (1997).

[8] C. Ferrie and R. Blume-Kohout, Maximum likelihood quantum state tomography is inadmissible, ArXiv:1808.01072 (2018).

[9] S. S. Tannu and M. K. Qureshi, in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19 (Association for Computing Machinery, New York, NY, USA, 2019), p. 987.

[10] G. Carleo and M. Troyer, Solving the quantum many-body problem with artificial neural networks, Science **355**, 602 (2017).

[11] D.-L. Deng, X. Li, and S. Das Sarma, Machine learning topological states, Phys. Rev. B **96**, 195145 (2017).

[12] J. C. Bridgeman and C. T. Chubb, Hand-waving and interpretive dance: An introductory course on tensor networks, J. Phys. A: Math. Theor. **50**, 223001 (2017).

[13] F. Verstraete and J. I. Cirac, Matrix product states represent ground states faithfully, Phys. Rev. B **73**, 094423 (2006).

[14] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, Neural-network quantum state tomography, Nat. Phys. **14**, 447 (2018).

[15] M. Cramer, M. B. Plenio, S. T. Flammia, R. Somma, D. Gross, S. D. Bartlett, O. Landon-Cardinal, D. Poulin, and Y.-K. Liu, Efficient quantum state tomography, Nat. Commun. **1**, 149 (2010).

[16] B. P. Lanyon, C. Maier, M. Holzäpfel, T. Baumgratz, C. Hempel, P. Jurcevic, I. Dhand, A. S. Buyskikh, A. J. Daley, M. Cramer, *et al.*, Efficient tomography of a quantum many-body system, Nat. Phys. **13**, 1158 (2017).

[17] J. Carrasquilla, G. Torlai, R. G. Melko, and L. Aolita, Reconstructing quantum states with generative models, Nat. Mach. Intell. **1**, 155 (2019).

[18] S. Ahmed, C. S. Muñoz, F. Nori, and A. F. Kockum, Quantum state tomography with conditional generative adversarial networks, ArXiv:2008.03240 (2020).

[19] G. E. Hinton, Training products of experts by minimizing contrastive divergence, Neural Comput. **14**, 1771 (2002).

[20] X. Gao and L.-M. Duan, Efficient representation of quantum many-body states with deep neural networks, Nat. Commun. **8**, 662 (2017).

[21] D.-L. Deng, X. Li, and S. Das Sarma, Quantum Entanglement in Neural Network States, Phys. Rev. X **7**, 021021 (2017).

[22] G. Torlai and R. G. Melko, Latent Space Purification Via Neural Density Operators, Phys. Rev. Lett. **120**, 240503 (2018).

[23] H.-G. Zimmermann, A. Minin, and V. Kusherbaeva, *ESANN 2011 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium)*, 27–29 April 2011, i6doc.com, Available at: http://www.i6doc.com/en/livre/?GCOI=28001100817300.

[24] S. Aaronson, The learnability of quantum states, Proc. R. Soc. A: Math. Phys. Eng. Sci. **463**, 3089 (2007).

[25] S. Aaronson, in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2018 (Association for Computing Machinery, New York, NY, USA, 2018), p. 325.

[26] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, ArXiv:1411.4028 (2014).

[27] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, Nature **549**, 242 (2017).

[28] C. Finn, P. Abbeel, and S. Levine, in *Proceedings of the 34th International Conference on Machine Learning—Volume 70*, ICML'17 (JMLR.org, Sydney, NSW, Australia, 2017), p. 1126.

[29] A. Elben, B. Vermersch, R. van Bijnen, C. Kokail, T. Brydges, C. Maier, M. K. Joshi, R. Blatt, C. F. Roos, and P. Zoller, Cross-Platform Verification of Intermediate Scale Quantum Devices, Phys. Rev. Lett. **124**, 010504 (2020).

[30] P. Smolensky, in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations* (MIT Press, Cambridge, MA, USA, 1986), p. 194.

[31] O. Breuleux, Y. Bengio, and P. Vincent, Quickly generating representative samples from an RBM-derived process, Neural Comput. **23**, 2058 (2011).

[32] Z. Jia, B. Yi, R. Zhai, Y. Wu, G. Guo, and G. Guo, Quantum neural network states: A brief review of methods and applications, Adv. Quantum Technol. **2**, 1800077 (2019).

[33] A. L. Maas, A. Y. Hannun, and A. Y. Ng, in *In ICML Workshop on Deep Learning for Audio, Speech and Language Processing* (Atlanta, GA, 2013).

[34] G. E. Hinton, in *Neural Networks: Tricks of the Trade*, edited by G. Montavon, G.B. Orr, K. R. Müller (Springer, Berlin, Heidelberg, 2012), vol. 7700, p. 599.

[35] D. Kingma and J. Ba, in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, (San Diego, CA, USA, 2015).

[36] J. Lever, M. Krzywinski, and N. Altman, Principal component analysis, Nat. Methods **14**, 641 (2017).

[37] H. Abraham, I. Y. Akhalwaya, G. Aleksandrowicz, T. Alexander, G. Alexandrowics, E. Arbel, A. Asfaw, C. Azaustre, P. B. AzizNgoueya, G. Barron, and L. Bello, Qiskit: An open-source framework for quantum computing (2019).

[38] J. Gray, Quimb: A python library for quantum information and many-body calculations, J. Open Source Softw. **3**, 819 (2018).

[39] M. Belkin, D. Hsu, S. Ma, and S. Mandal, Reconciling modern machine-learning practice and the classical bias–variance trade-off, Proc. Natl. Acad. Sci. **116**, 15849 (2019).

[40] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever, Deep double descent: Where bigger models and more data hurt, ArXiv:1912.02292 (2019).

[41] O. Higgott, D. Wang, and S. Brierley, Variational quantum computation of excited states, Quantum **3**, 156 (2019).