

## ARTICLE OPEN



# Adaptive characterization of spatially inhomogeneous fields and errors in qubit registers

Riddhi Swaroop Gupta<sup>1</sup> , Claire L. Edmunds<sup>1</sup>, Alistair R. Milne<sup>1</sup>, Cornelius Hempel<sup>1</sup> and Michael J. Biercuk<sup>1</sup>

New quantum computing architectures consider integrating qubits as sensors to provide actionable information useful for calibration or decoherence mitigation on neighboring data qubits, but little work has addressed how such schemes may be efficiently implemented in order to maximize information utilization. Techniques from classical estimation and dynamic control, suitably adapted to the strictures of quantum measurement, provide an opportunity to extract augmented hardware performance through automation of low-level characterization and control. In this work, we present an adaptive learning framework, Noise Mapping for Quantum Architectures (NMQA), for scheduling of sensor–qubit measurements and efficient spatial noise mapping (prior to actuation) across device architectures. Via a two-layer particle filter, NMQA receives binary measurements and determines regions within the architecture that share common noise processes; an adaptive controller then schedules future measurements to reduce map uncertainty. Numerical analysis and experiments on an array of trapped ytterbium ions demonstrate that NMQA outperforms brute-force mapping by up to 20× (3×) in simulations (experiments), calculated as a reduction in the number of measurements required to map a spatially inhomogeneous magnetic field with a target error metric. As an early adaptation of robotic control to quantum devices, this work opens up exciting new avenues in quantum computer science.

*npj Quantum Information* (2020)6:53; <https://doi.org/10.1038/s41534-020-0286-0>

## INTRODUCTION

Central to the scale-up of large-scale quantum computing systems will be the integration of automated procedures for hardware characterization, tune-up, and operation of realistic multi-qubit architectures<sup>1–6</sup>. This perspective is already validated in the literature; recent demonstrations of quantum supremacy subtly featured a graph-based learning routine in streamlining hardware calibration<sup>7</sup>. This successful demonstration motivates a broader exploration of how autonomous and adaptive learning can be deployed in addressing the ongoing challenges of calibration, as well as mitigation of decoherence and error in hardware.

Prior to the deployment of full quantum error correction, control solutions implemented as a form of “quantum firmware”<sup>8</sup> at the lowest level of the quantum computing software stack<sup>4,9</sup> provide an opportunity to improve hardware error rates using both open-loop dynamic error suppression<sup>10–15</sup> and closed-loop feedback stabilization. Closed-loop stabilization is a common low-level control technique for classical hardware<sup>16,17</sup>, but its translation to quantum devices faces challenges in the context of quantum-state collapse under projective measurement<sup>18,19</sup>.

One way feedback can be leveraged in quantum devices is through the adoption of an architectural approach embedding additional qubits as sensors at the physical level to provide actionable information on calibration errors and decoherence mechanisms<sup>20</sup>, e.g., real-time measurements of ambient field fluctuations<sup>21</sup>. In this so-called “spectator-qubit” paradigm, the objective is to spatially multiplex the complementary tasks of noise sensing and quantum data processing<sup>22</sup>. Fundamental to such an approach is the existence of spatial correlations in noise fields<sup>23</sup>, permitting information gained from a measurement on a spectator qubit used as an environmental or device-level hardware probe to be deployed in stabilizing proximal data qubits<sup>24</sup> used in the computation. Making the spectator-qubit paradigm practically useful in medium-scale architectures requires

characterization of the spatial variations in the processes inducing error—a process to which we refer as mapping—in order to determine which qubits may be actuated upon using information from a specific sensor. This is because spatial inhomogeneities in background fields can cause decorrelation between spectator and data qubits, such that feedback stabilization becomes ineffective or even detrimental. Given the relative “expense” of projective measurements—they are often the slowest operations in quantum hardware, and can be destructive, for example, the case of light leakage to neighboring devices—improving the efficiency of this mapping routine is of paramount importance.

In this paper, we introduce a new and broadly applicable framework for adaptive learning, denoted Noise Mapping for Quantum Architectures (NMQA), to efficiently learn about spatial variations in quantum computer hardware performance across a device through sampling, and apply this to the challenge of mapping an unknown “noise” field across a multi-qubit quantum device as a concrete example. NMQA is a classical filtering algorithm operated at the quantum firmware level, and is specifically designed to accommodate the nonlinear, discretized measurement model associated with projective measurements on individual qubits. The algorithm adaptively schedules measurements across a multi-qubit device, and shares classical state information between qubits to enable efficient hardware characterization, with specific emphasis on employing sampling approaches that reduce measurement overheads encountered in large architectures. We implement the NMQA framework via a two-layer particle filter and an adaptive real-time controller. Our algorithm iteratively builds a map of the underlying spatial variations in a specific hardware parameter across a multi-qubit architecture in real time by maximizing the information utility obtained from each physical measurement. This, in turn, enables the controller to adaptively determine the highest-value measurement to be performed in the following step. We evaluate the

<sup>1</sup>ARC Centre of Excellence for Engineered Quantum Systems, School of Physics, The University of Sydney, Sydney, NSW 2006, Australia. ✉email: riddhi.sw@gmail.com

performance of this framework to a noise-characterization task, where a decoherence-inducing field varies in space across a device. We study test cases by both numeric simulation on 1D and 2D qubit arrays, and application to real experimental data derived from Ramsey measurements on a 1D crystal of trapped ions. Our results demonstrate that NMQA outperforms brute-force measurement strategies by a reduction of up to 20× (3×) in the number of measurements required to estimate a noise map with a target error for simulations (experiments). These results hold for both 1D and 2D qubit regular arrays subject to different noise fields.

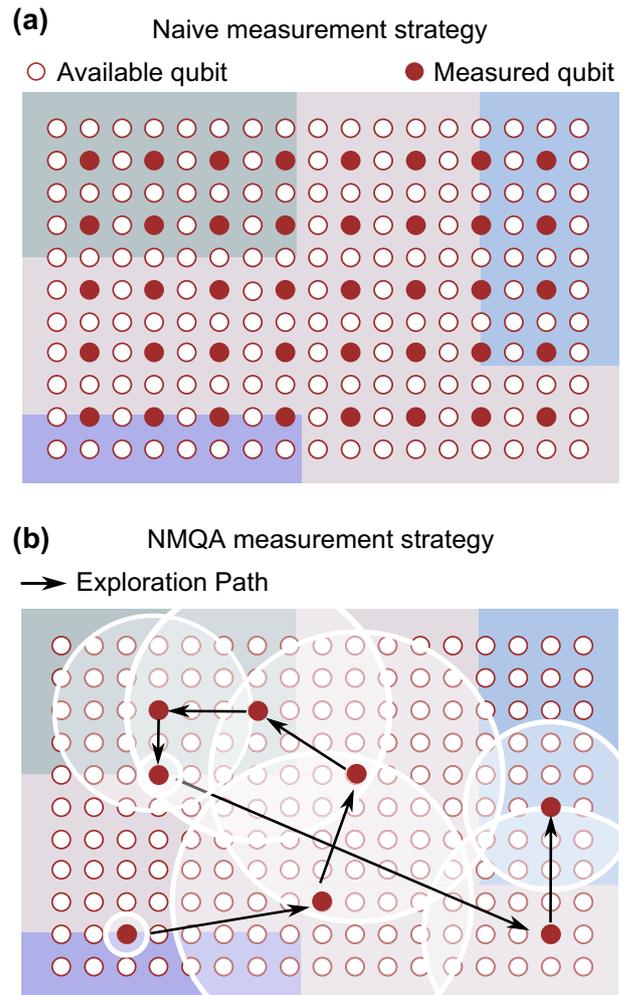
We consider a spatial arrangement of  $d$  qubits as determined by a particular choice of hardware. An unknown, classical field exhibiting spatial correlations extends over all qubits on our notional device, and corresponds to either intrinsic, spatially varying calibration errors, or extrinsic noise fields. Our objective is to build a map of the underlying spatial variation of the noise field with the fewest possible single-qubit measurements, performed sequentially. For concreteness, we conceive that each measurement is a single-shot Ramsey-like experiment in which the presence of the unknown field results in a measurable phase shift between a qubit's basis states at the end of a fixed interrogation period. This phase is not observed directly, but rather inferred from data, as it parameterizes the Born probability of observing a "0" or "1" outcome in a projective measurement on the qubit; our algorithms take these discretized binary measurement results as input data. The desired output at any given iteration  $t$  is a map of the noise field, denoted as a set of unknown qubit phases,  $F_t$ , inferred from the binary measurement record up to  $t$ .

The simplest approach to the mapping problem is to undertake a brute-force, "naive" strategy in which one uniformly measures sensor qubits across the array, depicted schematically in Fig. 1a. By extensively sampling qubit locations in space repeatedly, one can build a map of the underlying noise fields through the collection of large amounts of measurement information. Evidently, the naive, brute-force measurement approach is highly inefficient as it fails to recognize and exploit spatial correlations in the underlying noise field that may exist over a lengthscale that exceeds the inter-qubit spacing. This is a particular concern in circumstances where qubit measurements are a costly resource, e.g., in time, classical communication bandwidth, or qubit utilization when sensors are dynamically allocated.

The NMQA framework shown in Fig. 1b stands in stark contrast, taking inspiration from the problem of simultaneous localization and mapping in robotic control<sup>25–33</sup>. Here, the underlying spatial correlations of the noise are mapped using a learning procedure, which adaptively determines the location of the next, most-relevant measurement to be performed based on past observations. With this approach, NMQA reduces the overall resources required for spatial noise mapping by actively exploiting the spatial correlations to decide whether measurements on specific qubits provide an information gain.

In our algorithmic implementation, at every iteration  $t$ , information about the true map is contained in the state vector,  $X_t$ . Here, the word "state" refers to all quantities being inferred statistically from data, as opposed to an actual physical quantum state. Aside from information about the true map that we denote  $F_t \in X_t$ , the state vector,  $X_t$ , additionally contains information,  $R_t$ , which approximates spatial gradients on the true map. The probability distribution of  $X_t$  conditioned on data is called the posterior distribution, and it summarizes our best knowledge of the true map and its approximate spatial gradient information, given past measurement outcomes.

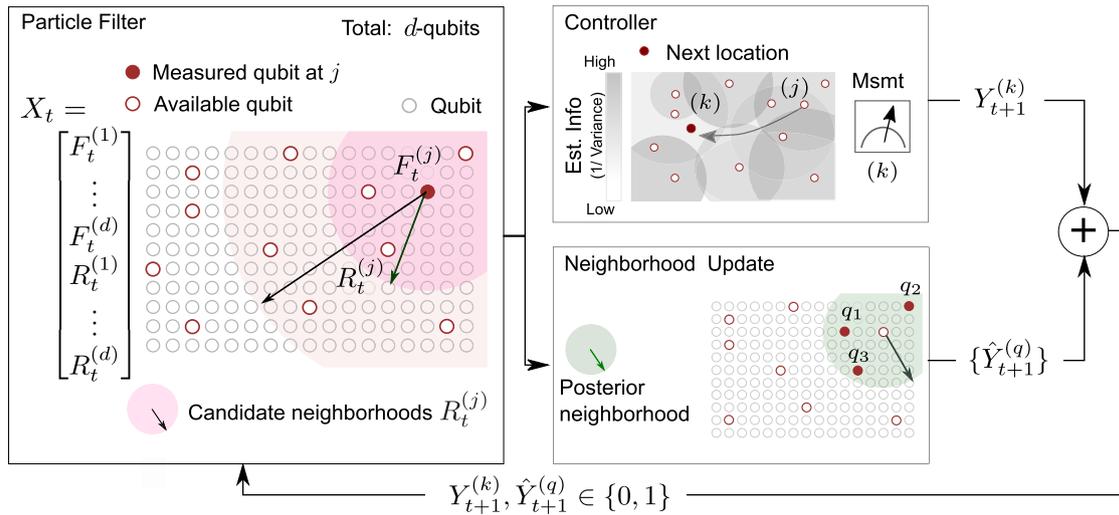
The key operating principle of NMQA is that we locally estimate the map value,  $F_t^{(j)}$  (a qubit phase shift with a value between  $[0, \pi]$  radians), before globally sharing the map information at the measured qubit  $j$  with neighboring qubits  $q \in Q_t$  in the vicinity of  $j$ . The algorithm is responsible for determining the appropriate size



**Fig. 1** Difference between a naive brute force and the NMQA measurement strategy in reconstructing an inhomogeneous background field. A spatial arrangement of qubits (red circles) is shown with a true unknown field with finite correlations (colored regions). **a** The naive strategy measures the field across the array using a regular grid (red filled circles). **b** The NMQA strategy iteratively chooses which qubit to measure next (black arrows), and additionally stores state-estimation information in a form shared across local neighborhoods (white shaded circles), which reflects the spatial characteristics of the underlying map.

of the neighborhood,  $Q_t$ , parameterized by the length-scale value,  $R_t^{(j)}$  (left panel, Fig. 2); accordingly, we may conceive  $R_t^{(j)}$  as the one-parameter mechanism to approximately learn local gradient information of the noise field. The combination of estimates of the state at each qubit and proxy local gradient information allows for a more efficient mapping procedure than brute-force sampling over all qubits.

In practice,  $Q_t$  eventually represents the set of qubits in a posterior neighborhood at  $j$  at the end of iteration  $t$ , and this set shrinks or grows as inference proceeds over many iterations. We are ignorant, a priori, of any knowledge of  $R_{t=0}^{(j)}$ . We set the prior distribution over lengthscales to be uniform over an interval  $[R_{\min}, R_{\max}]$  where initial estimates take values between  $R_{\min}$ , the smallest inter-qubit spacing, and  $R_{\max}$ , as a multiple of the maximal separation distance on the qubit array, in units of distance. In principle, any prior distribution over the lengthscales can be designed by practitioners. The collection of map values and lengthscales,  $X_t^{(j)} := \{F_t^{(j)}, R_t^{(j)}\}$ , at every qubit,  $j = 1, 2, \dots, d$ , is depicted as an extended-state vector.



**Fig. 2 Schematic overview of the NMQA iteration.** A particle filter estimates the map,  $F_t$ , and discovers neighborhoods (circular shaded) parameterized by  $R_t^{(j)}$  for sharing state information about site  $j$ . Posterior-state estimates from the particle filter are used by the controller to choose site  $k$  as the location of the next physical measurement,  $Y_{t+1}^{(k)}$ , based on regions of the highest estimated uncertainty (middle top). Meanwhile, posterior-state estimates are also used to share information at  $j$  within the posterior neighborhood,  $Q_t$  via data messages,  $\hat{Y}_{t+1}^{(q)}$ , between all neighboring qubits  $q \in Q_t$  (middle bottom) before commencing the next iteration. The input/output information loop is the outermost loop formed by a single physical measurement (in notation,  $Y$ ) and a set of data messages (in notation,  $\hat{Y}$ ). The arrow to site  $k$  from  $j$  does not relate to the transfer of any information, but rather, adaptive measurement scheduling by the controller to maximize information utility from the next physical measurement choice  $k$ .

We rely on an iterative maximum-likelihood procedure within each iteration of the algorithm to solve the NMQA inference problem, as the size of the state-space generally impedes analytic solutions even in classical settings<sup>32,34</sup>. In each iteration, we update  $F_t$  assuming that  $X_{t-1}$  is known, and subsequently  $R_t$  based on  $F_t$ . This structure is unique, and NMQA manipulates the joint probability distribution defined over  $F_t$  and  $R_t$ , such that we numerically implement an iterative maximum-likelihood procedure using techniques known as particle filters (see below). The final result of this first step in Fig. 2 is a posterior estimate of  $X_t$ , which represents our best knowledge given measurement data.

This estimate is now passed to an adaptive measurement scheduler, the NMQA controller, which attempts to maximize the information utility from each measurement. The NMQA controller adaptively selects the location  $k$  for the next physical measurement by choosing the region where posterior-state variance is maximally uncertain (Fig. 2, top-middle panel). The new measurement outcome, once collected, is denoted  $Y_{t+1}^{(k)}$  (a posterior-state variance is typically estimated using the properties of particles inside the filter<sup>35</sup>). Meanwhile, the posterior-state information at step  $t$  is used to generate data messages so that estimated map information can be shared with proximal neighboring qubits (Fig. 2, bottom-middle panel). The shared information is denoted by the set  $\{\hat{Y}_{t+1}^{(q)}, q \in Q_t\}$ , where the posterior information in  $Q_t$  at the end of  $t$  is equivalent to information in  $Q_{t+1}$  provided at the beginning of the next iteration,  $t+1$ . The data messages  $\{\hat{Y}_{t+1}^{(q)}\}$  are taken as an input to the algorithm in a manner similar to a physical measurement. Jointly, the new actual physical measurement,  $Y_{t+1}^{(k)}$ , and the set of shared information  $\{\hat{Y}_{t+1}^{(q)}\}$  form the inputs for the next iteration,  $t+1$ . Further technical information on the overall NMQA implementation is presented in “Methods”.

We approximate the theoretical nonlinear filtering problem embodied by NMQA using a class of techniques known as particle filters. In each iteration of the NMQA algorithm, we must update

information about the expected vectors  $F_t$  and  $R_t$  based on the latest measurement information acquired. This is a challenging data inference problem that shares commonality with a host of other inference problems in science and engineering (such as robotic mapping<sup>36–38</sup>) involving nonlinear, non-Gaussian properties or high-dimensional state spaces that are sparsely sampled. Particle filters are numerical approximation methods belonging to a general class of Sequential Monte Carlo algorithms used to approximately solve the classical Bayesian inference problems under such conditions<sup>39</sup>. The prolific use of particle filters stems from their success in nonlinear filtering applications in large state spaces of continuous random variables, relevant to the application considered here. In our application, the choice of a particle filter to implement core NMQA functionality accommodates the nonlinear measurement model associated with the discretized outcomes given by projective qubit readout, and large state spaces defined over all possible maps and associated lengthscales.

Like many inference methods, particle filters use Bayes rule to solve an inference problem. Under Bayesian inference, the desired solution is a posterior distribution representing the probability distribution of the inferred state conditioned on data. The defining feature of particle-filtering algorithms is that one sacrifices access to analytical expressions for the moments of a posterior distribution (e.g., the mean). Instead, the posterior distribution is approximately represented as a discrete collection of weighted “particles”. Each particle has two properties: a position and a weight. The position of the particle is effectively a hypothesis about the state  $X_t$ , i.e., a sample from the probability distribution for  $X_t$  given observed data. The weight specifies the likelihood or the importance of the particle in the estimation procedure. All weights are initially equal at step  $t=0$ , and after receiving measurement data at each step, the particles are “re-sampled”. This means that the original set of particles at step  $t$  is replaced by a set of “offspring” particles, where the probability that a parent is chosen to represent itself in the next iteration (with replacement) is directly proportional to its weight. Over many iterations, only the particles with the highest weights survive, and these surviving

particles form the estimate of the distribution of  $X_t$ , given data in our algorithm. At any  $t$ , the estimate of  $X_t$  can be obtained as the first moment of the posterior particle distribution, and similarly, true state uncertainty is estimated from the empirical variance of the particle distribution. The efficacy of any particle filter depends on how particle weights are computed, and many proposals to compute these weights exist. Further details on fundamental particle-filtering concepts and strategies to compute particle weights are introduced in refs. <sup>39,40</sup> and restated in section 2C of Supplementary Information.

We now provide a summary of the particle-filtering implementation used here, and highlight modifications that are unique to NMQA. In each iteration, we update  $F_t$  assuming that  $X_{t-1}$  is known, and subsequently  $R_t$  assuming  $F_t$ . This structure requires two particle types inside NMQA's particle filter:  $\alpha$  particles carry information about the full state vector,  $X_t$ , while  $\beta$  particles discover the optimal information-sharing neighborhood size,  $R_t^{(j)}$ , around qubit  $j$ . The two different types of particle sets are then used to manipulate the joint probability distribution defined over  $F_t$  and  $R_t$  to obtain an estimate of  $X_t$ , which represents our best knowledge, given measurement data.

Under the conditions articulated above, the NMQA filtering problem requires only two specifications: a prior or initial distribution for the state vector,  $X_0$ , at  $t=0$ , and a mechanism that assigns each particle with an appropriate weight based on measurement data. We take the approach where particle weights are computed using a specific type of likelihood function for measurement, and a novel branching process, whose convergence properties can be analyzed<sup>41</sup>. Assuming a uniform prior, we only need to define the global likelihood function incorporating both particle types. We label  $\alpha$  particles by the set of numbers  $\{1, 2, \dots, n_\alpha\}$ ; for each  $\alpha$  particle, we also associate a set of  $\beta$  particles denoted  $\beta^{(\alpha)}$ , with  $\beta$  taking values  $\{1, 2, \dots, n_\beta\}$ . Each  $\alpha$  particle is weighted or scored by a so-called likelihood function. This likelihood function is written in notation as  $g_1(\lambda_1, Y_t^{(j)})$ , where  $\lambda_1$  is a parameter of the NMQA model and  $Y_t^{(j)}$  makes explicit that only one physical measurement at one location is received per iteration. A single  $\beta^{(\alpha)}$  particle inherits the state from its  $\alpha$  parent, but additionally acquires a single, uniformly distributed sample for  $R_t^{(j)}$  from the length-scale prior distribution. The  $\beta$  particles are scored by a separate likelihood function,  $g_2(\lambda_2, Q_t)$ , where  $\lambda_2$  is another parameter of the NMQA model. Then the total likelihood for an  $(\alpha, \beta^{(\alpha)})$  pair is given by the product of the  $\alpha$ - and  $\beta$ -particle weights.

The functions  $g_1(\lambda_1, Y_t^{(j)})$  and  $g_2(\lambda_2, Q_t)$  are likelihood functions used to score particles inside the NMQA particle filter, and their mathematical definitions can be found in the Supplementary Methods. These functions are derived by representing the noise affecting the physical system via probability-density functions. The function  $g_1(\lambda_1, Y_t^{(j)})$  describes measurement noise on a local projective qubit measurement as a truncated or a quantized zero-mean Gaussian error model to specify the form of the noise-density function, with variance given by  $\Sigma_v$ . The function  $g_2(\lambda_2, Q_t)$  represents the probability density of "true errors" arising from choosing an incorrect set of neighborhoods while building the map. For each candidate map, the value of the field at the center is extrapolated over the neighborhood using a Gaussian kernel. Thus, what we call "true errors" are the differences between the values of the true, spatially continuous map and the NMQA estimate, consisting of a finite set of points on a map and their associated (potentially overlapping) Gaussian neighborhoods. We assume that these errors are truncated Gaussian distributions with potentially nonzero means  $\mu_F$  and variances  $\Sigma_F$  over many iterations, as detailed in the Supplementary Methods.

In general, the a priori theoretical design of the likelihood function and its parameters depends on the choice of the measurement procedure and physical application. For static field characterization, we impose a tightly regularized filtering

procedure by setting low variance values relative to the initial state vector  $X_{t=0}$  and machine precision. Low variance values correspond to physical assumptions that bit-flip errors are unlikely, and high-frequency temporal noise is averaged away for large  $T$ , by setting  $\Sigma_v = 10^{-4}$ . Further, we assume that the unknown field is continuous and enables an approximation via overlapping Gaussian neighborhoods, by setting  $\mu_F = 0$ ,  $\Sigma_F = 10^{-6}$ . As with any filtering problem, one may alternatively consider an optimization routine over design parameters and measurement protocols embodied by the likelihood functions, thereby removing all a priori knowledge of a physical application.

The two free parameters,  $\lambda_1, \lambda_2 \in [0, 1]$ , are used to numerically tune the performance of the NMQA particle filter. Practically,  $\lambda_1$  controls how shared information is aggregated when estimating the value of the map locally, and  $\lambda_2$  controls how neighborhoods expand or contract in size with each iteration. In addition, the numerically tuned values of  $\lambda_1, \lambda_2 \in [0, 1]$  provide an important test of whether or not the NMQA sharing mechanism is trivial in a given application. Nonzero values suggest that sharing information spatially actually improves performance more than just locally filtering observations for measurement noise. As  $\lambda_1, \lambda_2 \rightarrow 1$ , the set  $\{\hat{Y}_{t+1}^{(q)}\}$ , is treated as if they were the outcomes of real physical measurements by the algorithm. However, in the limit  $\lambda_1, \lambda_2 \rightarrow 0$ , no useful information sharing in space occurs, and NMQA effectively reduces to the naive measurement strategy.

The information-sharing mechanism described above also influences how the controller schedules a measurement at the next qubit location. At each  $t$ , the particle filter can access the empirical mean and variance of the posterior distribution represented by the particles. In particular, the particle filter stores estimated posterior variance of the  $\beta$  particles at qubit location  $j$ , if this location was physically measured during the filtration until  $t$ . The ratio of the empirical variance of the  $\beta$  particles to the posterior estimate of  $R_t^{(j)}$ , is then compared for all qubit locations. The controller selects a location  $k$  uniformly randomly between locations with the largest values of this ratio, corresponding to a location with the highest uncertainty in neighborhood lengthscales. In the absence of information sharing (setting  $\lambda_1, \lambda_2 = 0$ ), the controller effectively uniformly randomly samples qubit locations, and the difference between NMQA and a deterministic brute-force measurement strategy is reduced to the random vs. ordered sampling of qubit locations.

Detailed derivations for mathematical objects and computations associated with NMQA, and an analysis of their properties using standard nonlinear filtering theory are provided in ref. <sup>41</sup>.

## RESULTS

Application of the NMQA algorithm using both numerical simulations and real experimental data demonstrates the capabilities of this routine for a range of spatial arrangements of  $d$  qubits in 1D or 2D. For the results reported in this section, NMQA receives a single "0" or "1" projective measurement,  $Y_t^{(j)}$ , in each iteration  $t$  and location  $j$ . We have also validated numerically that benefits persist with continuous (classically averaged) measurement outcomes, where many single-qubit projective measurements,  $\{Y_t^{(j)}\}$ , are averaged in each  $t$  and  $j$  in Fig. 5 of Supplementary Figures.

Our evaluation procedure begins with the identification of a suitable metric for characterizing mapping performance and efficiency. We choose a Structural Similarity Index (SSIM)<sup>42</sup>, frequently used to compare images in machine-learning analysis. This metric compares the structural similarity between two images, and is defined mathematically in "Methods". It is found to be sensitive to improvements in the quality of images while giving robustness against, e.g., large single-pixel errors that frequently plague norm-based metrics for vectorized images<sup>43,44</sup>.

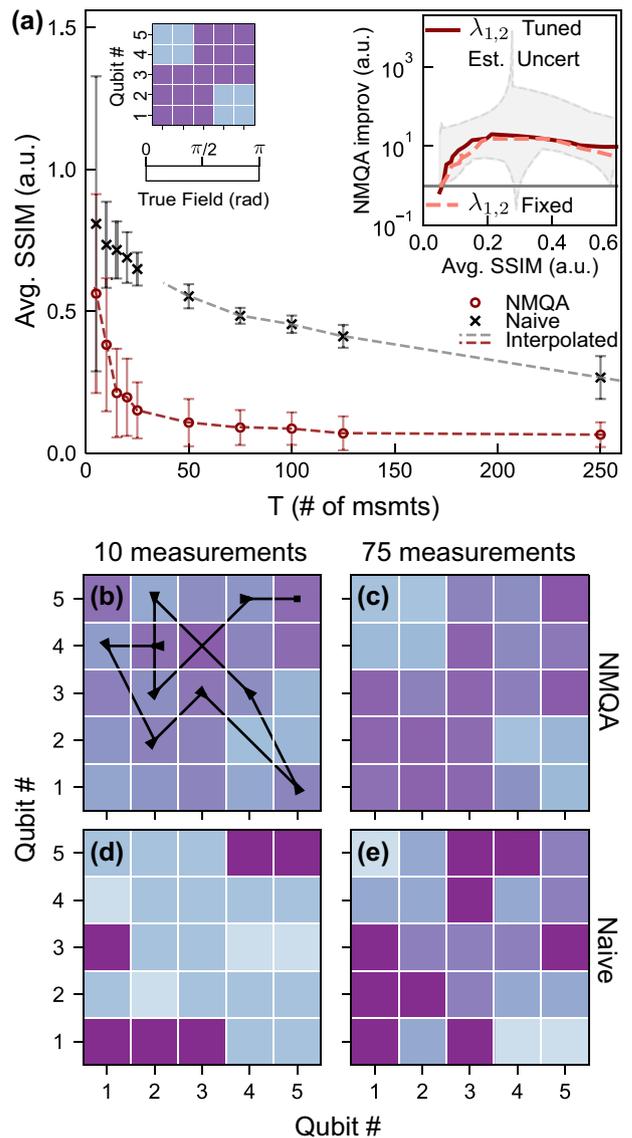
In our approach, we compare the true map and its algorithmic reconstruction by calculating the SSIM; a score of zero corresponds to ideal performance, and implies that the true map and its reconstruction are identical.

We start with a challenging simulated example, in which  $d = 25$  qubits are arranged in a 2D grid. The true field is partitioned into square regions with relatively low and high values (Fig. 3a, left inset) to provide a clear structure for the mapping procedure to identify. In this case, the discontinuous change of the field values in space means that NMQA will not be able to define a low-error candidate neighborhood for any qubits on the boundary. Both NMQA and naive are executed over a maximum of  $T$  iterations, such that  $t \in [1, T]$ . For simplicity, in most cases, we pick values of  $T \geq d$  as multiples of the number of qubits,  $d$ , such that every qubit is measured the same integer number of times in the naive approach, which we use as the baseline in our comparison. If  $T < d$ , the naive approach uniformly randomly samples qubit locations. Both NMQA and the naive approach terminate when  $t = T$ , and a single-run SSIM score is calculated using the estimated map for each algorithm in comparison with the known underlying “true” noise map. The average value of the score over 50 trials is reported as Avg. SSIM, and error bars represent 1 standard deviation (s.d.) for all data figures.

The Avg. SSIM score as a function of  $T$  is shown in the main panel of Fig. 3a. For any choice of  $T$ , NMQA reconstructs the true field with a lower SSIM than the naive approach, indicating better performance. The SSIM score also drops rapidly with  $T$  when using NMQA, compared with a more gradual decline in the naive case. Increasing  $T \gg d$  leads to not only an improvement in the performance of both algorithms, but also a convergence of the scores as every qubit is likely to be measured multiple times for the largest values considered here. Representative maps for  $T = 10$  and  $T = 75$  measurements under both the naive and NMQA mapping approaches are shown in Fig. 3b–e, along with a representative adaptive measurement sequence employed in NMQA in panel b. In both cases, NMQA provides estimates of the map that are closer to the true field values, whereas naive maps are dominated by estimates at the extreme ends of the range, which is characteristic for simple reconstructions using sparse measurements.

It is instructive to represent these data in a form that shows the effective performance enhancement of the NMQA mapping procedure relative to the naive approach; the upper-right inset in Fig. 3a reports the improvement in the reduction of the number of measurements required to reach a desired Avg. SSIM value. We perform linear interpolation for each raw dataset in Fig. 3a, depicted using dashed lines in the main panel. The ratio of the interpolated data as a function of Avg. SSIM value is shown in the upper-right inset of Fig. 3a (red solid). The interpolation process is repeated to obtain the difference between the maximum upper and minimum lower error trajectories (shaded gray) using error bars on underlying raw data for these results.

The shape of the performance improvement curve is linked to the total amount of information provided to both algorithms. At high Avg. SSIM scores on the far right of the figure, both naive and NMQA algorithms receive very few measurements, and map reconstructions are dominated by errors. In the intermediate regime, a broad peak indicates that NMQA outperforms brute-force measurement in reducing the total number of qubit measurements over a range of moderate-to-low Avg. SSIM scores for map reconstruction error, up to  $7\times$  for Avg. SSIM = 0.1. Similar performance is achieved for a range of other qubit-array geometries and characteristics of the underlying field (Supplementary Figures).



**Fig. 3 Performance evaluation of NMQA vs. naive using simulated 2D qubit arrays.** **a** Left inset: 2D array of 25 qubits with “square” field shown as a colorscale. Main panel: Avg. SSIM score vs. total measurement budget  $T$  for NMQA (open circles) and naive (crosses) averaged over 50 trials, error bars represent 1 s.d. Dashed lines linearly interpolate the data. Upper-right inset: ratio of naive to NMQA measurements vs. Avg. SSIM score using tuned  $\lambda_1, \lambda_2$  (solid line) at each  $T$ ; fixed choice  $\lambda_1 = 0.89, \lambda_2 = 0.97$  (dashed line) tuned for  $T = 20$ . For tuned  $\lambda_{1,2}$ , the maximum achieved ratio and its corresponding error score are (19.8, 0.211); the ratio drops below unity at Avg. SSIM = 0.06. The shaded gray region shows the maximum upper and minimum lower error for tuned  $\lambda_{1,2}$ . Interpolation occurs for Avg. SSIM = [0.05, 0.6] to exclude small  $T$  data with large error bars. Sharp features in est. uncertainty arise from computing max-upper or least-lower bounds using numerical ratios of underlying data. **b–e** Columns show single-run maps using a total number of  $T = 10$  or  $T = 75$  measurements plotted for the NMQA (**b, c**) and naive approach (**d, e**), with a representative control path shown in (**b**). The numerically tuned parameters are  $(\lambda_1, \lambda_2) = (0.89, 0.97)$  and  $(0.93, 0.68)$  for **b, c**, respectively. The single-map SSIM values are (**b**) 0.37, (**c**) 0.09, **d** 0.8, and **e** 0.47. Other parameters:  $d = 25$ ;  $T = 5, 10, 15, 20, 25, 50, 75, 100, 125, 250$ ;  $\Sigma_v = 10^{-4}$ ,  $\Sigma_f = 10^{-6}$ ; true low and high field values of  $0.25\pi$  and  $0.75\pi$  radians.

Near the origin at Avg. SSIM  $< 0.05$ , an extremely large number of measurements are required to achieve low Avg. SSIM scores. For  $T \rightarrow \infty$  and when all locations are measured, we theoretically expect the relative performance ratio to approach unity for any true field. In empirical analysis, the relative improvement of NMQA to naive is based on projections of the interpolation procedure rather than raw data. Our simulations at  $T = 250$  are being used to extrapolate at very low SSIM scores where the implied  $T \sim 10^3$  for both naive and NMQA; consequently, the resulting ratio is very sensitive to a change in parameters of the interpolation procedure. An analytical, rather than empirical, investigation of NMQA's convergence at low-error scores is presented in ref. <sup>41</sup>.

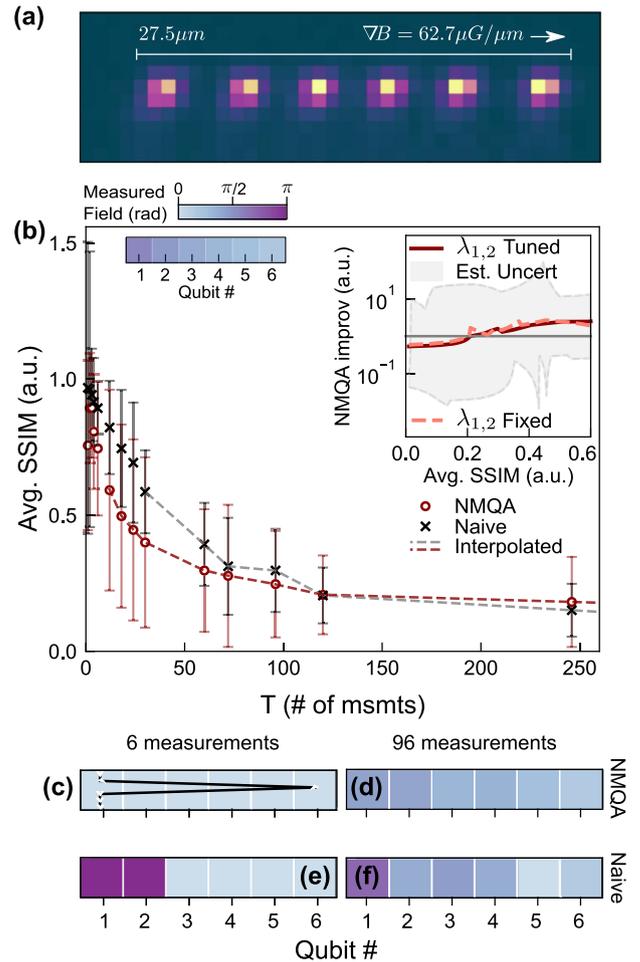
All of our results rely on appropriate tuning of the NMQA particle filter via its parameters  $\lambda_1$  and  $\lambda_2$ . Numerical tuning of these parameters is conducted for each  $T$ , and these data are represented as a solid red curve in Fig. 3a. We also demonstrate that using fixed values for these parameters only marginally degrades performance, as indicated by the dashed line in the upper-right inset of Fig. 3a.

We now apply the NMQA mapping algorithm to real experimental measurements on qubit arrays. In Fig. 4, we analyze Ramsey experiments conducted on an array of  $^{171}\text{Yb}^+$  ions confined in a linear Paul trap, with trap frequencies  $\omega_{x,y,z}/2\pi \approx (1.6, 1.5, 0.5)$  MHz. Qubits are encoded in the  $^2S_{1/2}$  ground-state manifold, where we associate the atomic hyperfine states  $|F = 0, m_F = 0\rangle$  and  $|F = 1, m_F = 0\rangle$  with the qubit states  $|0\rangle$  and  $|1\rangle$ , respectively. State initialization to  $|0\rangle$  via optical pumping and state detection are performed using a laser resonant with the  $^2S_{1/2} - ^2P_{1/2}$  transition near 369.5 nm. Laser-induced fluorescence (corresponding to projections into state  $|1\rangle$ ) is recorded using a spatially resolving EMCCD camera yielding simultaneous readout of all ions. In this experiment, qubit manipulation is carried out using microwave radiation at 12.6 GHz delivered via an in-vacuum antenna to all qubits globally.

The trapped ions experience an uncontrolled (“intrinsic”), linear magnetic field gradient that produces spatially inhomogeneous qubit frequencies over the trap volume. When manipulated using a global microwave control field, this results in a differential phase accumulation between qubits. The magnitude of the gradient is illustrated in Fig. 4a, superimposed on an image of six ions in the bright state  $|1\rangle$ , and is not observed to drift or vary on any timescale relevant to these experiments. We aim to probe this field gradient through the resulting qubit detuning and associated differential phase accumulation throughout each measurement repetition. In this case, NMQA may be thought of as either an adaptive noise-mapping or calibration routine.

A preliminary Ramsey experiment, in which the interrogation time is varied, and fringes are observed, confirms that at an interrogation time of 40 ms, the accumulated relative phase in a Ramsey measurement is  $< \pi$  radians. A total of 25, 500 Ramsey measurements with a wait time of 40 ms are performed on all six ions in parallel. For each repetition, a standard machine-learning image classification algorithm assigns a “0” (dark) or “1” (bright) to each ion based on a previously recorded set of training data. From averaging over repetitions, we construct a map of the accumulated phase (and hence the local magnetic field inducing this phase) on each qubit, shown schematically in the left inset of Fig. 4b. We consider the field extracted from this standard averaging procedure over all repetitions of the Ramsey experiment, at each ion location, as the “true” field against which mapping estimates are compared using the SSIM score as introduced above.

We employ the full set of  $6 \times 25,500$  measurements as a data bank on which to evaluate and benchmark both algorithms. At each iteration, the algorithm determines a measurement location in the array, and then randomly draws a single, discretized measurement outcome (0 or 1) for that ion from the data bank. The rest of the algorithmic implementation proceeds as before. Accordingly, we expect that the naive approach must smoothly



**Fig. 4 Performance evaluation of NMQA vs. naive using experimental data.** **a** Image of six trapped  $^{171}\text{Yb}^+$  ions in the fluorescing  $|1\rangle$  state averaged over multiple exposures of  $750\ \mu\text{s}$ . **b** Left inset: colorscale indicating a phase shift induced by a magnetic field gradient across a 1D array of six qubits. Main panel: Avg. SSIM score vs. total measurement budget  $T$  for NMQA (open circles) and naive (crosses) algorithms averaged over 50 trials, error bars represent 1 s. **d**. Dashed lines linearly interpolate data. Upper-right inset: ratio of naive to NMQA measurements vs. Avg. SSIM score for tuned  $\lambda_1, \lambda_2$  (solid line) at each  $T$ , and fixed choice  $(\lambda_1, \lambda_2) = (0.95, 0.99)$  (dashed line) tuned for  $T = 24$ . For tuned  $\lambda_{1,2}$ , the maximum ratio and the corresponding error score are  $(2.7, 0.6)$ ; the ratio drops below unity at Avg. SSIM = 0.18. Shaded gray shows maximum upper and minimum lower empirical error for tuned  $\lambda_{1,2}$ . Sharp features in est. uncertainty arise from computing max-upper or least-lower bounds using numerical ratios of underlying data. **c–f** Columns show single-run maps using a total number of  $T = 6$  or  $T = 96$  measurements plotted for the NMQA (**c, d**) and naive (**e, f**), with a representative control path shown in **c**. The numerically tuned parameters are  $(\lambda_1, \lambda_2) = (0.97, 0.94)$  for **c** and  $(0.97, 0.95)$  for **d**. Single-map SSIM values are (**c**) 0.99, **d** 0.35, **e** 0.77, and **f** 0.50. Other parameters:  $d = 6$ ;  $T = 1, 2, 3, 4, 6, 12, 18, 24, 30, 60, 72, 96, 120, 246$ ;  $\Sigma_v = 10^{-4}$ ,  $\Sigma_F = 10^{-6}$ .

approach a SSIM score of zero as  $T$  increases (Fig. 4b). In these experiments, we experience a large measurement error arising from the necessary detection time of  $750\ \mu\text{s}$  associated with the relatively low quantum efficiency and the effective operating speed of the EMCCD camera. Compensating this via extension of the measurement period leads to an asymmetric bias due to state decays that occur during the measurement process in  $\approx 1.5$  ms ( $|1\rangle \rightarrow |0\rangle$ ) and  $\approx 30$  ms ( $|0\rangle \rightarrow |1\rangle$ ) under the laser power and quantization magnetic field strength used in our experiment. In

our implementation, neither NMQA nor the brute-force algorithm was modified to account for this asymmetric bias in the detection procedure, although in principle, the image classification algorithm employed to determine qubit states from fluorescence detection can be expanded to account for this. Therefore, both NMQA and the naive approach are affected by the same detection errors. Despite this complication, we again find that NMQA outperforms the naive mapping algorithm by a factor of 2–3 in the number of required measurements (Fig. 4b), with expected behavior at the extremal values of Avg. SSIM.

## DISCUSSION

In this work, we presented NMQA—a framework for adaptive learning where we reconstruct spatially varying parameters associated with qubit hardware, derived from autonomously and adaptively scheduled measurements on a multi-qubit device. We developed an iterative, maximum-likelihood procedure implemented via a two-layer particle filter to share state-estimation information between qubits within small spatial neighborhoods, via a mapping of the underlying spatial correlations of the parameter being measured. An adaptive controller schedules future measurements in order to reduce the estimated uncertainty of the map reconstruction. We focused on the specific application of mapping an unknown decoherence-inducing noise field, but this methodology could equally be applied to the characterization of nearly any device parameter subject to spatial inhomogeneities, and where efficiency in measurement sampling is valued. This could include, for instance, calibration of qubit Rabi rates due to nonuniform coupling of devices to their control lines, or variations in qubit frequencies arising from device-fabrication tolerances.

Numerical simulations and calculations run on real experimental data demonstrated that NMQA outperforms a naive mapping procedure by reducing the required number of measurements up to  $3\times$  in achieving a target map similarity score on a 1D array of six trapped ytterbium ions (up to  $20\times$  in a 2D grid of 25 qubits using simulated data). Beyond these example demonstrations, the key numerical evidence for the correctness of NMQA's functionality came from the observation that the tuned values of  $\lambda_1, \lambda_2 \gg 0$  for all results reported here. Since numerically tuned values for these parameters were found to be nonzero, we conclude that information sharing in NMQA is nontrivial, and the algorithm departs substantially from a brute-force measurement strategy. This contributes to the demonstrated improvements in Avg. SSIM scores when using NMQA in Figs. 3 and 4.

Overall, achieving suitable adaptive learning through NMQA requires that the  $(\lambda_1, \lambda_2)$  parameters are appropriately tuned using measurement datasets as part of a training procedure. To select  $(\lambda_1, \lambda_2)$ , we chose a  $(\lambda_1, \lambda_2)$  pair with the lowest expected error. In practice, one cannot calculate the true error as the true field is unknown. Additional numerical analysis shows that tuned  $\lambda_1, \lambda_2 > 0$  improves NMQA performance, and that this performance improvement gradually increases from the bottom-left to the top-right corner of the unit square away from the case  $\lambda_1 = \lambda_2 = 0$  (Fig. 1; Supplementary Figures) for a range of different noise fields with regularly spaced 1D or 2D configurations. This numerically observed gradual improvement in performance as a function of  $\lambda_{1,2} \rightarrow 1$  suggests that theoretical approaches to deducing “optimal” regions for  $(\lambda_1, \lambda_2)$  may be possible. Meanwhile, in a specific application, a practitioner may only have access to the real-time rate of change of state estimates and residuals, i.e., a particle filter generates a predicted measurement at location  $j$ , which can be compared with the next actual measurement received at  $j$ . Monitoring the real-time rate of change of state estimates and/or residuals in a single run can be used to develop effective stopping criteria to set  $(\lambda_1, \lambda_2)$  with minimal a priori knowledge of the physical system. If, for example, total preparation and measurement time is on the order of tens or hundreds of

microseconds, then an experiment of  $T \sim 10^5$  measurements takes a few minutes to complete. For ion traps, the relevant timescales for temporal drift in control fields are on the order of hours. Meanwhile, run times for a single NMQA iteration are on the order of 1 ms in Python under typical operating conditions. Thus, it is possible to numerically tune  $\lambda_1, \lambda_2$  for NMQA using small  $T$  measurements before proceeding with a desired experiment. Future work to speed up and integrate NMQA into low-level experimental control software appears promising as particle filters do not require complicated linear algebra operations (e.g., matrix inversions), and enable rapid, parallel computations across particles. The particle-filtering structure of NMQA presents as-yet unexplored opportunities in software execution to explore in-cycle tuning with run times on par with physical measurements. These extensions to the operability of NMQA are active research areas.

The framework we have introduced is flexible and can accommodate temporal variations in the system, such as drifts in the map and changes in the availability of sensor qubits. Without a priori information about dynamics, the state-transition kernel is set to identity, and in this form, our assumption is that NMQA accommodates low-frequency drift quasi-statically. Temporal dynamics rapid compared with the measurement cycle are generally averaged over, and ignored by the algorithm. In intermediate regimes, it may be possible to accommodate fluctuations as measurement noise or error in our knowledge of a qubit's given location. A formal spatiotemporal analysis with NMQA is the subject of future work, and combines a nonidentity state-transition kernel with NMQA's unique branching process.

We have thus far presented a comparison of NMQA with a naive, brute-force measurement strategy, but it is also possible to consider the error-scaling behavior under NMQA against a strengthened competing measurement procedure. For instance, it is possible to perform many measurements at each qubit location in order to accurately (within a user-defined bound) establish the relevant mapped value for that qubit. One may gradually increase the number of simultaneous measurements at a single location  $j$  as 2, 3, ..., 50, ... measurements per iteration  $t$  and so on, which means that NMQA takes in a classically post-processed measurement result rather than a “0” or “1” projective measurement input. For a true field of Fig. 3, our investigations show that in the region of Avg. SSIM score  $\approx 0.2$ , NMQA outperforms naive by a factor greater than  $\gg 2$  using 2–15 measurements per  $t$ , and is above unity in the regions 15–50 measurements per  $t$  (Fig. 5; Supplementary Figures). In this case, determining where to measure next in order to build the map still benefits from estimating length-scales to approximate local field gradient information; minimizing uncertainty in these estimated lengthscales informs an adaptive strategy.

In the limit that many qubit measurements are taken at every location (such as performing many averaged measurements in parallel), the naive brute-force measurement strategy essentially transitions from an inference to an interpolation problem—i.e., given near-perfect knowledge of a finite-point set, we seek to approximately reconstruct an unknown continuous function in bivariate parameter space. In this regime, the arrangement of qubits on a 2D plane affects the accuracy of the resulting reconstruction. One approach to this problem is to restrict physical measurements to a subset of qubits within the register, requiring that the value of an unknown, continuous, spatially varying field must be inferred at the remaining set of unmeasured qubits. The study of this approach, and comparison of the error-scaling behavior of an optimally constructed classical interpolant vs. NMQA is the subject of future work<sup>45</sup>. We are also excited to explore how exploitation of hardware platforms in which qubit locations are not rigidly fixed during fabrication, such as with trapped ions, may allow sublattice spatial resolution by dynamically translating individual qubits during the mapping procedure. Our work is part of an exciting new area of future study exploring

the intersection between hardware architectures and control solutions<sup>8,41</sup> in NISQ-era quantum computers, and may also have bearing on distributed networks of quantum sensors.

## METHODS

We summarize the structure of the NMQA algorithm in Algorithm 1 using a pseudocode representation. The first part of the algorithm consists of an initialization procedure, which ensures that all particles are sampled from the prior for extended-state vector at  $t=0$ , giving  $X_0$ . All particles are equally weighted before measurement data are received. For  $t > 0$ , the function `PropagateStates` represents the transition-probability distribution for Markov  $X_t$ , i.e., it represents identity dynamics, and is a placeholder for future extensions to model dynamical  $F_t$ . In each  $t$ , a single physical measurement is received. This triggers a set of update rules for  $F_t$  and  $R_t$ . We note that the state variables,  $F_t$  and  $R_t$ , are updated in a specific order within each timestep  $t$ . The order of these computations corresponds to the iterative maximum-likelihood approximation for the NMQA framework.

### Algorithm 1. NMQA

```

procedure NMQA ( $d$  qubit locations,  $\lambda_1, \lambda_2$ )
  if  $t = 0$  then
    procedure INITIALIZE ( $X_0$ )
      for  $a \in \{1, 2, \dots, n_a\}$  do
        Initially sample  $X_0^{(a)} \sim \pi_0$ 
        Initially compute  $W_0^{(a)} = \frac{1}{n_a}$ 
      end for
    end procedure
  end if
  while  $1 \leq t < T$  do
    if Controller then
       $j_t, Y_t^{(j_t)} \leftarrow \text{CONTROLLER}(X_{t-1}) \triangleright$  Qubit  $j$ , at  $t$ 
    end if
    for  $a \in \{1, 2, \dots, n_a\}$  do
       $\{X_t^{(a)}\} \leftarrow \text{PROPAGATESTATES}(\{X_{t-1}^{(a)}\})$ 
      Update  $F_t^{(\cdot),(a)}$  via  $\{Y_t^{(j_t)}, \{\hat{Y}_t^{(q_t)}\}, \lambda_1\}$ 
       $\{X_t, W_t\}^{(\alpha, \beta_a)} \leftarrow \text{COMPUTEWEIGHTS}(\{X_t^{(a)}\})$ 
       $\{X_t^{(\alpha, \beta_a)}, \frac{1}{n_a n_{\beta}}\} \leftarrow \text{RESAMPLE}(\{X_t, W_t\}^{(\alpha, \beta_a)})$ 
      Update  $R_t^{(j_t),(a)}$ 
       $\{X_t, W_t\}^{(a)} \leftarrow \text{COLLAPSE}\beta(\{X_t^{(\alpha, \beta_a)}, \frac{1}{n_a n_{\beta}}\})$ 
       $\{X_t^{(a)}, \frac{1}{n_a}\} \leftarrow \text{RESAMPLE}(\{X_t, W_t\}^{(a)})$ 
    end for
     $\{\hat{Y}_{t+1}^{(q)}\}_{q \in Q_{t+1}} \leftarrow \text{GENERATE}\hat{Y}(\text{Posterior } X_t)$ 
  end while
end procedure
function COMPUTEWEIGHTS( $\{X_t^{(a)}\}$ )
  for  $a \in \{1, 2, \dots, n_a\}$  do
    Compute  $\tilde{W}_t^{(a)} = g_1(\lambda_1, Y_t^{(j)})$ 
     $\{X_t^{(\alpha, \beta_a)}\} \leftarrow$  Generate  $\beta$ -layer
  for  $\beta_a \in \{1, 2, \dots, n_{\beta}\}$  do
    Compute  $\tilde{W}_t^{(\beta_a|a)} = g_2(\lambda_2, Q_t)$ 
  end for
  Normalize  $\tilde{W}_t^{(\beta_a|a)}$ 
end for
  Normalize  $\tilde{W}_t^{(a)}$ 
  Compute  $W_t^{(\alpha, \beta_a)} = \tilde{W}_t^{(\beta_a|a)} \tilde{W}_t^{(a)} \forall \alpha, \{\beta_a\}$ 
  Return  $n_a n_{\beta}$  particles and weights  $\{X_t, W_t\}^{(\alpha, \beta_a)}$ 
end function

```

For each type of particle, the weights are computed according to NMQA likelihood functions  $g_1(\lambda_1, Y_t^{(j)})$  and  $g_2(\lambda_2, Q_t)$ . These likelihood calculations are local about each qubit location, enabling NMQA to receive multiple measurements in either the same or different locations in a single iteration  $t$ . Multiple measurements at one location improve the local map estimate  $F_t^{(j)}$  at  $j$  via  $g_1(\lambda_1, \cdot)$  before the estimated  $F_t^{(j)}$  is shared with proximal qubits. Multiple measurements at different but proximal locations improve the estimation of lengthscales via  $g_2(\lambda_2, Q_t)$ . This means that the net effect of parallelized measurements also depends on how these measurements sample an unknown field. We restrict our current scope to a single physical measurement per iteration, and the effect of spatial sampling on functional reconstruction is the scope of a forthcoming paper.

As typical of a standard particle filter, NMQA also supports parallelization by ensuring that state transformations and weight calculations for each particle occur independently of other particles. Dynamical evolution and the computation of  $\alpha$ - and  $\beta$ -particle weights are parallelizable for both  $\alpha$  and  $\beta$  particles in software. While the normalization of particle weights represents a scalar rescaling of all particles in a layer, this is computationally inexpensive. In our Python implementation of NMQA, a computational cost in sorting particles during resampling is incurred, and alternative resampling techniques have been investigated in particle-filtering literature<sup>36,37,46</sup>. Nevertheless, the key strength of particle-filtering techniques is that they enable parallel computations while avoiding matrix inversions that can be computationally expensive or ill-conditioned in numerical inference procedures, for example, Gaussian Process Regression<sup>47</sup> or Kalman Filtering techniques<sup>48</sup>. NMQA benefits from these advantages of standard particle filters.

A detailed description of the nonlinear filtering problem represented by NMQA and its convergence properties is provided in ref.<sup>41</sup>. The correctness and convergence of typical particle filters are explored in the literature<sup>40,49</sup>, and we discuss it specifically for NMQA<sup>41</sup>.

### Structural similarity metric definition

For all analyses, we use a risk metric called the Structural Similarity Index (SSIM)<sup>42</sup>. This metric is used to conduct optimization of NMQA parameters, and assess performance relative to the naive measurement strategy. For two vectorized images  $x$  and  $y$ , the metric is defined as

$$s(x, y) = \frac{(2\mu_x \mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (1)$$

$$\text{SSIM}(x, y) := |1 - s(x, y)| \quad (2)$$

In the formula above,  $\mu_i, \sigma_i^2, i = x, y$  represent the sample estimates of the means and variances of the respective vectorized images, and  $\sigma_{xy}$  captures correlation between images. The term  $s(x, y)$  is the key metric developed in ref.<sup>42</sup>, and it includes arbitrary constants  $C_1 = C_2 = 0.01$ , which stabilize the metric for images with means or variances close to zero. The ideal score given by  $s(x, y)$  is unity, and corresponds uniquely to the case  $x = y$ . We report the absolute value of the deviations from the ideal score of unity, where the direction of the deviation is ignored as given by  $\text{SSIM}(x, y)$ . For our application, this  $\text{SSIM}(x, y)$  metric lies between  $[0, 1]$  (negative expected values of  $s(x, y)$  are not seen in our numerical demonstrations). We report the average of  $\text{SSIM}(x, y)$  values over 50 trials as Avg. SSIM.

### DATA AVAILABILITY

Access to the codebase and data required to reproduce all figures is provided via <https://github.com/qcl-sydney/nmqa> without restrictions.

Received: 29 October 2019; Accepted: 21 May 2020;

Published online: 12 June 2020

### REFERENCES

1. Yao, N. Y. et al. Scalable architecture for a room temperature solid-state quantum information processor. *Nat. Commun.* **3**, 800 (2012).
2. Monroe, C. et al. Large-scale modular quantum-computer architecture with atomic memory and photonic interconnects. *Phys. Rev. A* **89**, 022317 (2014).
3. Veldhorst, M., Eenink, H. G. J., Yang, C. H. & Dzurak, A. S. Silicon CMOS architecture for a spin-based quantum computer. *Nat. Commun.* **8**, 1766 (2017).

4. Jones, N. C. Layered architecture for quantum computing. *Phys. Rev. X* **2**, 031007 (2012).
5. Kielpinski, D., Monroe, C. & Wineland, D. J. Architecture for a large-scale ion-trap quantum computer. *Nature* **417**, 709 (2002).
6. Franke, D. P., Clarke, J. S., Vandersypen, L. M. K. & Veldhorst, M. Rentas rule and extensibility in quantum computing. *Microprocess. Microsys.* **67**, 1–7 (2019).
7. Arute, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).
8. Ball, H., Nguyen, T., Leong, P. H. W. & Biercuk, M. J. Functional basis for efficient physical layer classical control in quantum processors. *Phys. Rev. Appl.* **6**, 064009 (2016).
9. Carvalho, A. et al. Software tools for quantum control: Improving quantum computer performance through noise and error suppression. Preprint at <https://arxiv.org/abs/2001.04060> (2020).
10. Brown, K. R., Aram, A. W. & Chuang, I. L. Arbitrarily accurate composite pulse sequences. *Phys. Rev. A* **70**, 052318 (2004).
11. Merrill, J. T. & Brown, K. R. Progress in compensating pulse sequences for quantum computation. In *Quantum Information and Computation for Chemistry* (ed. Kais, S.) 241–294 (John Wiley, Sons, Inc., 2014).
12. Khodjasteh, K. & Viola, L. Dynamically error-corrected gates for universal quantum computation. *Phys. Rev. Lett.* **102**, 080501 (2009).
13. Soare, A. et al. Experimental noise filtering by quantum control. *Nat. Phys.* **10**, 825–829 (2014).
14. Paz-Silva, G. A. & Viola, L. General transfer-function approach to noise filtering in open-loop quantum control. *Phys. Rev. Lett.* **113**, 250501 (2014).
15. Kabytayev, C. et al. Robustness of composite pulses to time-dependent control noise. *Phys. Rev. A* **90**, 012316 (2014).
16. Gelb, A. *Applied Optimal Estimation* (MIT Press, 1974).
17. Landau, I. D., Lozano, R., M'Saad, M. & Karimi, A. *Adaptive Control: Algorithms, Analysis and Applications* (Springer Science, Business Media, 2011).
18. Mavadia, S., Frey, V., Sastrawan, J., Dona, S. & Biercuk, M. J. Prediction and real-time compensation of qubit decoherence via machine learning. *Nat. Commun.* **8**, 14106 EP (2017).
19. Gupta, R. S. & Biercuk, M. J. Machine learning for predictive estimation of qubit dynamics subject to dephasing. *Phys. Rev. Appl.* **9**, 064042 (2018).
20. Majumder, S., de Castro, L. A. & Brown, K. R. Real-time calibration with spectator qubits. *npj Quantum Inf.* **6**, 1–9 (2020).
21. Cooper, A. et al. Time-resolved magnetic sensing with electronic spins in diamond. *Nat. Commun.* **5**, 3141, <https://doi.org/10.1038/ncomms4141> (2014).
22. Brown, K. R., Kim, J. & Monroe, C. Co-designing a scalable quantum computer with trapped atomic ions. *npj Quantum Inf.* **2**, 16034, <https://doi.org/10.1038/npjqi.2016.34> (2016).
23. Postler, L. et al. Experimental quantification of spatial correlations in quantum dynamics. *Quantum* **2**, 90 (2018).
24. Hirose, M. & Cappellaro, P. Coherent feedback control of a single qubit in diamond. *Nature* **532**, 77–80 (2016).
25. Cadena, C. et al. Past, present, and future of simultaneous localization and mapping: toward the robust-perception age. *IEEE Transac. Robot.* **32**, 1309–1332 (2016).
26. Bergman, N. Recursive bayesian estimation. *Department of Electrical Engineering, Linköping University, Linköping Studies in Science and Technology. Doctoral Dissertation* **579**, 11 (1999).
27. Stachniss, C. et al. Particle filters for robot navigation. *Found. Trends® Robot.* **3**, 211–282 (2014).
28. Durrant-Whyte, H. & Bailey, T. Simultaneous localization and mapping: part I. *IEEE Robot. Autom. Mag.* **13**, 99–110 (2006).
29. Bailey, T. & Durrant-Whyte, H. Simultaneous localization and mapping (SLAM): part II. *IEEE Robot. Autom. Mag.* **13**, 108–117 (2006).
30. Murphy, K. P. Bayesian map learning in dynamic environments. In *Advances in Neural Information Processing Systems* (eds Solla, S. A., Leen, T. K. & Müller, K.) 1015–1021. <https://papers.nips.cc/book/advances-in-neural-information-processing-systems-12-1999> (Neural Information Processing Systems, 2000).
31. Andrew, H. Multi-robot simultaneous localization and mapping using particle filters. *Int. J. Robot. Res.* **25**, 1243–1256 (2006).
32. Thrun, S., Burgard, W. & Fox, D. *Probabilistic Robotics* (MIT Press, 2005).
33. Thrun, S., Burgard, W. & Fox, D. A probabilistic approach to concurrent mapping and localization for mobile robots. *Auton. Robots* **5**, 253–271 (1998).
34. Thrun, S. A probabilistic on-line mapping algorithm for teams of mobile robots. *Int. J. Robotics Res.* **20**, 335–363 (2001).
35. Bain, A. & Crisan, D. *Fundamentals of Stochastic Filtering*, Stochastic Modelling and Applied Probability (Springer, 2009).
36. Beevers, K. R. & Huang, W. H. Fixed-lag sampling strategies for particle filtering SLAM. In *Proceedings 2007 IEEE International Conference on Robotics and Automation* 2433–2438. <https://www.tib.eu/en/search/id/TIBKAT%3A555477428/Proceedings-Roma-Italy-10-14-April-2007/> (IEEE, 2007).
37. Grisetti, G., Stachniss, C. & Burgard, W. Improving grid-based slam with Rao-Blackwellized particle filters by adaptive proposals and selective resampling. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* 2432–2437. <https://doi.org/10.1109/ROBOT.2005.1570477> (IEEE, Barcelona, Spain, 2005).
38. Poterjoy, J. A localized particle filter for high-dimensional nonlinear systems. *Monthly Weather Rev.* **144**, 59–76 (2016).
39. Doucet, A., De Freitas, N. & Gordon, N. In *Sequential Monte Carlo Methods in Practice* (eds Doucet, A., De Freitas, N. & Gordon, N.) 3–14 (Springer, 2001).
40. Bain, A. & Crisan, D. *Fundamentals of Stochastic Filtering*, Vol. 60 (Springer Science, Business Media, 2008).
41. Gupta, R. S. & Biercuk, M. J. Convergence analysis for autonomous adaptive learning applied to quantum architectures. Preprint at <https://arxiv.org/abs/1911.05752> (2019).
42. Wang, Z., Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE T. Image Process.* **13**, 600–612 (2004).
43. Chen, Y., Garcia, E. K., Gupta, M. R., Rahimi, A. & Cazzanti, L. Similarity-based classification: concepts and algorithms. *J. Machine Learn. Res.* **10**, 747–776 (2009).
44. Wang, Z. & Bovik, A. C. Mean squared error: love it or leave it? a new look at signal fidelity measures. *IEEE Signal Process. Mag.* **26**, 98–117 (2009).
45. Gupta, R. S., Govia, L. C. & Biercuk, M. J. Integration of spectator qubits into quantum computer architectures for hardware tuneup and calibration. Preprint at <https://arxiv.org/abs/2004.13059> (2020).
46. Li, T., Bolic, M. & Djuric, P. M. Resampling methods for particle filtering: classification, implementation, and strategies. *IEEE Signal Process. Mag.* **32**, 70–86 (2015).
47. Williams, C. K. & Rasmussen, C. E. *Gaussian Processes for Machine Learning*, Vol. 2, No. 3 (MIT Press, Cambridge, MA, 2006).
48. Grewal, M. S. *Kalman Filtering* (Springer, 2011).
49. Crisan, D. & Doucet, A. A survey of convergence results on particle filtering methods for practitioners. *IEEE T. Signal Process.* **50**, 736–746 (2002).

## ACKNOWLEDGEMENTS

The authors thank V.M. Frey for proposing methods for single-ion-state detection using camera images, and S. Sukkarieh, A.C. Doherty, and M. Hush for useful discussions. This work was partially supported by the ARC Centre of Excellence for Engineered Quantum Systems CE170100009, the US Army Research Office under Contract W911NF-12-R-0012, and a private grant from H. & A. Harley.

## AUTHOR CONTRIBUTIONS

The NMQA theoretical framework and numerical implementations were devised by R.S.G. based on research directions set by M.J.B. R.S.G. and M.J.B. co-wrote the paper. A.R.M., C.L.E., and C.H. led all experimental efforts and contributed to the paper draft.

## COMPETING INTERESTS

The authors declare no competing interests.

## ADDITIONAL INFORMATION

**Supplementary information** is available for this paper at <https://doi.org/10.1038/s41534-020-0286-0>.

**Correspondence** and requests for materials should be addressed to R.S.G.

**Reprints and permission information** is available at <http://www.nature.com/reprints>

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



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative

Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly

from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2020