Integration of spectator qubits into quantum computer architectures for hardware tune-up and calibration

Riddhi Swaroop Gupta [1,*] Luke C. G. Govia [2] and Michael J. Biercuk [1]

[1]*ARC Centre of Excellence for Engineered Quantum Systems, School of Physics, The University of Sydney, New South Wales 2006, Australia*
[2]*Raytheon BBN Technologies, 10 Moulton St., Cambridge, Massachusetts 02138, USA*

Performing efficient quantum computer tune-up and calibration is essential for growth in system complexity. In this work we explore the link between facilitating such capabilities and the underlying architecture of the physical hardware. We focus on the specific challenge of measuring ("mapping") spatially inhomogeneous quasistatic calibration errors using spectator qubits dedicated to the task of sensing and calibration. We introduce an architectural concept for such spectator qubits: arranging them spatially according to prescriptions from optimal two-dimensional approximation theory. We show that this insight allows for efficient reconstruction of inhomogeneities in qubit calibration, focusing on the specific example of frequency errors which may arise from fabrication variances or ambient magnetic fields. Our results demonstrate that optimal interpolation techniques display near optimal error scaling in cases where the measured characteristic (here the qubit frequency) varies smoothly, and we probe the limits of these benefits as a function of measurement uncertainty. For more complex spatial variations, we demonstrate that the noise mapping for quantum architectures formalism for adaptive measurement and noise filtering outperforms optimal interpolation techniques in isolation and, crucially, can be combined with insights from optimal interpolation theory to produce a general purpose protocol.

I. INTRODUCTION

The scale-up of near-term quantum devices presents new challenges associated with fabrication, tune-up, characterization, and control as the total number of qubits increases [1–5]. For instance, in larger devices, we simultaneously see that fabrication tolerances may lead to performance variation across systems [6], while the number of measurements required in order to bring systems online grows rapidly. These concerns add to the general issue of decoherence due to undesired coupling of qubits to spatially inhomogeneous ambient fields. Accordingly, the procedure by which quantum devices must be characterized and tuned-up presents a challenge which grows with the size and complexity of each new generation of device [7,8].

In response, there has been an emergence of interest in the deployment of automated and nondestructive data inference and control approaches to this problem, exploiting the presence of spatial correlations in the performance variations [8]. One interesting approach to this problem involves characterization of classical noise or device variations in space using spectator qubits [9]; additional dedicated devices are employed to gain information about hardware performance such that information-carrying data qubits may operate undisturbed. Even within this framework we are still faced with the challenge of efficiently extracting information from potentially large numbers of these ancillary sensing devices.

In such circumstances, classical adaptive filtering techniques can play an essential role in efficiently characterizing, calibrating, and controlling mesoscale quantum devices in an attempt to extract improved performance [2,4,10]. In earlier work, the authors presented two distinct methods with radically different applications for the approximate reconstruction of unknown continuous physical phenomena using discretized projective measurements on qubits. The first method [11] was an efficient approach to quantum state tomography of single-mode continuous variable systems using bivariate optimal Lagrange interpolation at the Padua points [12–16]. The second method was inspired by probabilistic robotics and enabled adaptive scheduling of measurements on two-dimensional (2D) multiqubit arrays for noise spatial mapping, referred to as noise mapping for quantum architectures (NMQA) [17,18].

In this work, we combine these apparently disparate techniques in order to address the problem of how to most efficiently extract useful information for mesoscale tune-up and calibration using spectator qubits. We model the underlying hardware performance variations as a "field" and thus seek the most efficient means to characterize the spatial field. To this end we are inspired by the observation that ideal interpolation theory provides insights on how to conduct optimal sampling on any continuously varying, bivariate function (e.g., space, time, and relative phase). We associate such optimal sampling points with the physical locations of sensor qubits in order to efficiently determine spatial variations in the field at the location of the (unmeasured) data qubits.

In particular, we suggest sensor qubits should be located on a grid defined by the so-called Padua points [13–16],

which enables efficient Lagrange polynomial interpolation of the underlying field. We employ numeric simulations of field characterization using different algorithmic approaches to estimation and explore the influence of the underlying sensor-qubit locations on the quality of the estimates. We demonstrate that Lagrange-Padua reconstructions in 2D outperforms other interpolation or statistical estimation methods for the reconstruction of polynomial fields, while using $\sim 10\times$ fewer sensor qubits. Our simulations show how these benefits are lost in conditions when optimal interpolation theory cannot be applied, and demonstrate that the alternative adaptive measurement approach NMQA [17,18] achieves the lowest expected error. We study the impact of quantum projection noise on field estimation quality, including the role of averaging over the discretized outcomes of projective measurements. Our results demonstrate that NMQA is agnostic to the spatial arrangement of sensor qubits on hardware, motivating the use of a Padua sensor grid as a default architectural choice for embedded sensors.

Our work shows that using sensors on a Padua grid is always the favored strategy. In cases where the field is well approximated by a polynomial, interpolation methods may be used; in cases where this is violated, NMQA may be employed without degradation of the mapping. Ultimately combining insights from the estimation procedure into the physical layout of devices is a key finding of this work. We outline our approach in Sec. II, where we present a classical spatial reconstruction and prediction problem over a 2D arrangement of sensor and data qubits. Two distinct types of device configurations are considered where a sensor-qubit sublattice is nested in a fixed data-qubit grid. For each device configuration, we introduce the appropriate spatial reconstruction strategy and test performance. Results from numeric simulations and comparative analyses are presented in Sec. III. We conclude with a summary and brief future outlook in Sec. IV.

II. APPROACHES TO SPATIAL FIELD ESTIMATION VIA SPECTATOR QUBITS

We consider a notional 2D multiqubit array in which qubits are partitioned into "sensor" and "data" qubits; these noninteracting qubits are subject to spatial inhomogeneities in a device parameter (e.g., qubit frequency), modeled as an external classical field with finite spatial correlations. For concreteness we implement a term coupling to the qubit Hamiltonian $\propto \hat{\sigma}_z$ Pauli operator, as would be the case for qubit frequency variations or an inhomogeneous magnetic field. The question arising in this work is the extent to which the spatial configuration of the sensor qubits affects the accuracy of the field reconstruction at the proximal data qubits.

To probe this Hamiltonian term we assume a single-shot Ramsey experiment with a fixed interrogation time τ can be performed on each qubit, though the general framework is compatible with arbitrary measurement routines. Each measurement yields a "0" or "1" outcome representing the state of the qubit. By considering a sensor-qubit state initialized in a superposition of Pauli $\hat{\sigma}_z$ eigenstates in the appropriate interaction picture, an arbitrary detuning $\delta\omega(t)$ of the sensor-qubit frequency with respect to the system master clock is

expressible for each independent sensor-qubit Hamiltonian as

$$H_{\text{int}}(t) := \frac{\delta\omega(t)}{2}\hat{\sigma}_z, \tag{1}$$

where the detuning $\delta\omega(t)$ is written in a general time-dependent form. A nonzero detuning term gives rise to an experimentally measurable phase $\phi(\tau)$ between the qubit superposition states through the application of two temporally separated pulses in a Ramsey experiment [19],

$$\phi(\tau) = (\phi_2 - \phi_1) + \int_0^\tau \delta\omega(t')dt', \tag{2}$$

where τ is the temporal separation in which the qubit undergoes free evolution and $\phi_{1,2}$ are the reference phases of each applied control pulse. The accumulated phase $\phi(\tau)$ can be seen as a path difference leading to interference between the probability amplitudes of qubit basis states as $|0\rangle + e^{-i\phi(\tau)}|1\rangle$. The probability of obtaining a particular 0 or 1 outcome varies as

$$\Pr(Y) := \cos[\phi(\tau)/2]^Y \sin[\phi(\tau)/2]^{1-Y}, \quad Y \in \{0, 1\}, \tag{3}$$

where Y is a qubit-state projective measurement outcome and the expression above represents the ideal case in the absence of any additional measurement noise. Restricting to the static case where $\delta\omega(t) \equiv \delta\omega, \forall t$, we fix τ and consider repeated Ramsey measurements on sensor qubits distributed on a 2D multiqubit array. Here, $\phi(\tau)$ for each sensor qubit exhibits spatial variations over 2D space, accruing either to hardware control imperfections or the presence of inhomogeneous external fields. The choice of τ determines the resolution at which detuning errors are resolved for identifying the frequencies of qubits that may be far off resonance during calibration and tune-up.

Our procedure yields space-stamped discrete binary samples at the location of sensor qubits reflecting the underlying spatial inhomogeneities of the dephasing field. In the limit of high-fidelity measurements on each sensor qubit, the task of mapping the field at the data-qubit locations transitions from an estimation problem to a bivariate interpolation or an approximation problem. Let x (similarly, x') correspond to the spatial coordinates of a sensor qubit (data qubit) in 2D, and let $f(x)$ [similarly, $f(x')$] be the estimated field value measured by repeated projective measurements. A key insight in this work is the recognition that some interpolation techniques are associated with an optimal choice of point set at which to sample, $\chi := \{x_i\}_{i=1}^{|\chi|}$, and that these points *should* be associated with the spatial locations of sensor qubits. The canonical example in 1D is the Chebshev points. In 2D, the Padua points are a recently discovered family of point sets for optimally sampling a bivariate continuous function, for which Lagrange polynomial interpolation yields the lowest-error reconstruction [13–16].

The Padua points are families of point sets that occur on the unit square $[-1, 1]^2$. The geometric interpretation of Padua points can be obtained by envisioning them as equally spaced points along a generating curve on the unit square, defined by

$$\gamma_\kappa(z) := (-\cos[(\kappa + 1)z], -\cos(\kappa z)), \tag{4}$$

where z is any real continuous parameter. The intersection of this generating curve with itself, the edges of the square, or

its vertices yields the Padua point set. The Padua points are equispaced along z and indexed by j, j' as

$$z_{(j,j')} := \frac{j\kappa + j'(\kappa + 1)}{\kappa(\kappa + 1)}\pi, \quad j, j' \geqslant 0, \quad j + j' \leqslant \kappa. \quad (5)$$

Each Padua point set is associated with a positive integer *order*, κ, which indicates that a Lagrange interpolant can be constructed at the Padua points using a family of polynomials with degree $n \leqslant \kappa$. Different κ's yield Padua point sets, χ_κ, that differ in both size and arrangement on the unit square, with the number of points in the set given by

$$|\chi_\kappa| = \frac{(\kappa + 2)(\kappa + 1)}{2}. \quad (6)$$

The structure of the Padua grid is unique with respect to the data-qubit grid for $\kappa > 2$, while for $\kappa = 1, 2$ the Padua grid is degenerate with a square grid containing the relevant number of sites.

For any point in the unit square $x' \in [-1, 1]^2$, the Padua-Lagrange interpolant [13,14] of a true field, denoted f, is written:

$$\mathcal{L}(f)(x') := \sum_{x \in \chi_\kappa} f(x)l(x, x'). \quad (7)$$

Here the Lagrange basis polynomials, $l(x, x')$, form an orthonormal basis over the Padua points, such that for any two Padua points $x_1, x_2 \in \chi_\kappa$, $l(x_1, x_2) = 1$ if and only if $x_1 = x_2$, and $l(x_1, x_2) = 0$ otherwise [14,15]. If either one or both of x_1, x_2 is not a Padua point, $l(x_1, x_2) \geqslant 0$, and can be computed efficiently from the coordinate locations x_1 and x_2 using matrices involving the so-called "product grids" of the Chebychev polynomials in 1D, as detailed in [5], and restated for completeness in the Appendixes.

Locating sensor qubits at Padua points may be contrasted with methods in which the sensor-qubit array is matched to the underlying geometry of the data qubits, but incorporated as a nested, regular sublattice. In this circumstance, a polynomial basis need not be an optimal choice for interpolation. We may instead perform interpolation using radial basis functions (RBF) [20,21]. The RBF method uses a nonpolynomial (e.g., sigmoidal or Gaussian) function basis to conduct interpolation and scales well to multivariate interpolation problems.

So far we have discussed deterministic analysis methods such as interpolation, but we also consider nonlinear statistical estimation techniques that incorporate a projective-measurement model. We choose an algorithm presented in previous literature as NMQA [17,18], which estimates an unknown 2D field by adaptively scheduling measurements on sensor qubits. NMQA can be implemented directly on any sensor-qubit arrangement, including both regular (rectangular) lattices or Padua locations for sensor qubits. However, not all of the available sensor qubits need to be measured, as NMQA's controller selects the next physical measurement adaptively based on state estimation in the previous iteration step. It is thus distinct from interpolation methods as it focuses on building a "map" of f iteratively as new measurements are obtained using a spatial information sharing and prediction mechanism. NMQA procedures are characterized in detail in Refs. [17,18].

III. RESULTS AND DISCUSSION

In this section, we employ a numeric simulation to characterize the performance of these techniques for true fields exhibiting different forms of spatial variation, while incorporating different spatial arrangement of sensors on hardware. Our studies consider a fixed 5×5 regular (square) grid of data qubits with an overlaid grid of sensor qubits arranged at locations dictated by the chosen estimation scheme [Fig. 1(a)]. Simulated measurements from the sensor qubits are used to reconstruct a map of the field f at the locations of the data qubits on the array. For any mapping algorithm we calculate error via the uniform error, i.e., the infinity norm, and refer to this quantity as the "mapping error." The data inference procedures discussed here are compared against a baseline approach using a square grid of sensors and simply assigning sensor results to nearest-neighbor data qubits ("standard" protocol). This standard protocol uses a total of nine sensor qubits for a 5×5 grid data-qubit grid, where each sensor qubit is associated with up to four nearest-neighboring data qubits. Full details of the specific realization of the standard protocol grid in four possible orientations and all other hardware geometries used in comparative simulations are provided in the Appendixes.

A. Dependence of interpolation performance on sensor grid

We begin by examining the dependence of mapping error on the choice of interpolation strategy and sensor grid in Fig. 1(b). Calculations take as input data the binary outcomes 0 or 1 of simulated projective single-qubit measurements and receive the same fixed total measurement budget. Measurements are "batch processed" and the total measurement budget is fixed as a multiple of the grid size, so that every sensor qubit is measured m times. Thus, for $m > 1$, measurements at every sensor qubit are used to produce an estimate for the expectation value of the measurement, and this floating-point number is then given to the algorithm.

The main panel in Fig. 1(b) shows simulated mapping error vs the polynomial order n of different fields f, as achieved using different interpolation methods. Each data point in Fig. 1(b) shows the average mapping error and standard deviation for a set of 50 randomly sampled polynomial fields of a given order n. Examples of true fields for different n are depicted as a color scale in the top row of insets. The mapping error for Lagrange-Padua interpolation is plotted for $\kappa = n$ and $m = 50$ measurements per sensor qubit (black dashed triangles). This data set may be compared directly with a RBF interpolation using the maximum number of sensors possible irrespective of the order n of the field f: a 54 sensor-qubit regular grid (green dashed) with an approximately $2:1$ ratio of sensor to data qubits.

Since the number of sensors in Lagrange-Padua interpolation scales with n, we may also directly compare interpolation on a regular vs Padua grid with approximately the same number of sensor qubits. The choices of $d \times d$ regular grid for $d = 2, 3, 4, 5, 9$ and Padua grid with $\kappa = 1, 3, 4, 6, 9$ form different spatial geometries with approximately the same number of sensor qubits. In these specific cases, we plot the output of RBF interpolation as a separate data set (blue squares),
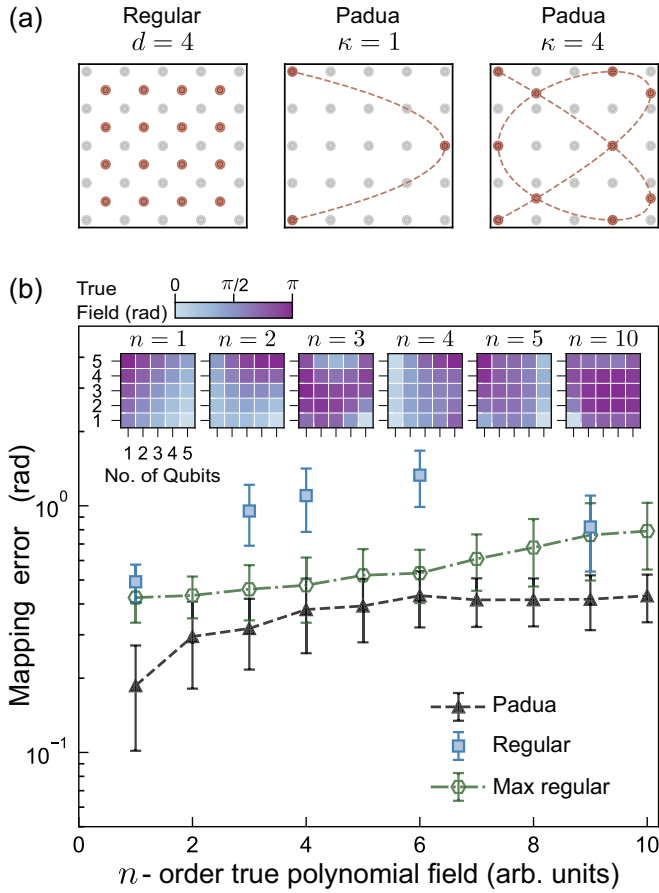
FIG. 1. Impact of sensor-qubit layout on field mapping quality in a 2D device. (a) Fixed 5×5 data qubits (gray dots) used in all analyses. Left to right: sensor qubits (red dots) are arranged in regularly spaced $d \times d$ grid for $d = 4$ (left) or according to Padua points of order $\kappa = 1, 4$ (middle, right). Generating curve for Padua point sets illustrated by red dashed line. (b) Mapping error in radians (rad) vs polynomial order n for three different interpolation strategies. Top row of insets depicts an example of true polynomial field of degree n on a fixed 5×5 regular grid of data qubits. Main panel: each data marker corresponds to an average over 50 trials of randomly generated true fields. In a single trial, a true field is generated as a polynomial of degree n with random coefficients, and the output of the polynomial is scaled and shifted to lie between $[0, \pi]$ radians. Interpolation at Padua points of order κ is performed using Lagrange polynomials with $\kappa = n$ (black dashed triangles). Interpolation on a $d \times d$ regular grid is performed using RBF, where the total number of sensors is within ± 3 qubits on both regular and Padua grids, with $d = 2, 3, 4, 5, 9$ (blue squares). For $d = 9$, sensor qubits that overlap with data qubits are removed, leading to a total of 54 sensor qubits. In the extreme case where the number of sensor qubits on regular grid are much greater than Padua grid for all orders $n < 9$, RBF interpolation is performed on a fixed, regular grid using $d = 9$ ("Max regular" in green dashed markers). Error bars represent one standard deviation within the distribution of 50 random trial polynomials. For all interpolation schemes $m = 50$ measurements per sensor qubit is used.

where blue data markers for polynomial order $n < 5$ ($n \geqslant 5$) correspond to a maximum size difference of ± 1 (± 3) sensor qubits between regular vs Padua grids.

In Fig. 1(b) we observe that, for all polynomial orders n, the Lagrange-Padua interpolation approach outperforms both alternative RBF interpolation strategies, demonstrating the impact of the sensor-qubit architecture on interpolation quality. This observation holds irrespective of whether regular and Padua grids are approximately the same size or if RBF is provided a distinct advantage with a high number of sensor qubits.

The inclusion of error in sensor-qubit measurements due to quantum projection noise represents a deviation from the core assumptions of ideal interpolation theory (for both RBF and Padua), where the value of the field at the sensors is assumed to be known exactly. Indeed, comparisons to optimal interpolation theory are only formally facilitated in the regime where ideal measurements are obtained as $m \to \infty$. Nonetheless, Lagrange-Padua reconstructions give low-error reconstructions of polynomial fields for $\kappa = n$ despite the fact that m is finite and that all interpolation methods receive noisy measurements at the location of sensor qubits.

B. Comparison between interpolation and adaptive measurement strategies

Having confirmed the utility of the Lagrange-Padua approach, we now relax assumptions of ideal interpolation theory in Fig. 2. We plot average mapping error vs the total number of sensor qubits, but we change the true field so that a representation on the finite polynomial basis increases in difficulty from (a) to (c), which have fixed underlying fields. For each data point, we simulate 50 repetitions of the measurement outcomes for the same underlying field and report an average mapping error and standard error.

Panel (a) in Fig. 2 corresponds to a simple linear variation, and in panel (b) we plot a common test function in interpolation literature known as the Franke function [22]. This function consists of a weak linear background favoring a Lagrange-Padua approach, but with the addition of a superposition of two Gaussian functions, favoring the Gaussian basis in RBF interpolation. In panel (c), the true field is nonpolynomial with a form $\propto \cos[\exp(2x + y)] \sin(y)$ that does not permit any finite polynomial or Gaussian representation.

The deviation from fields exactly describable by polynomial functions motivates the inclusion of the alternate adaptive mapping strategy, NMQA. In NMQA, a single-qubit binary measurement is fed iteratively into the algorithm; NMQA revises the state estimation procedure after each measurement and the NMQA controller iteration adaptively chooses which sensor qubit to measure next. This adaptive measurement procedure means that, for any fixed total measurement budget, sensor qubits under NMQA are not all uniformly measured m times.

We compare the different interpolation and adaptive measurement strategies across these three different fields using simulations setting $m = 50$, but varying the number of sensor qubits employed. Figure 2(a) numerically demonstrates that the Lagrange-Padua interpolation method for low order κ performs the best of our chosen methods. This benefit arises because the underlying field is a degree-1 polynomial. In this case, the simple structure of the underlying field makes the use of additional sensor qubits with $\kappa > 1$ unnecessary, as
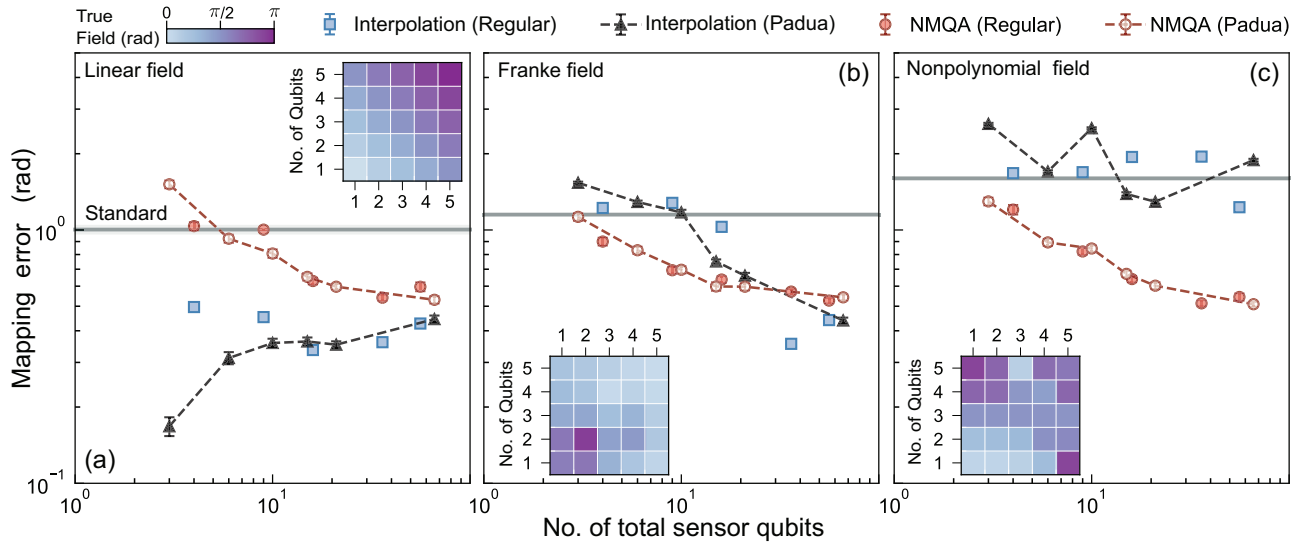
FIG. 2. Performance of interpolation strategies and NMQA for reconstructing different true fields for both regular and Padua geometries. True fields are depicted as a color scale on a fixed 5×5 grid of data qubits in insets. Main panels (a)–(c) show expected mapping error in radians (rad) vs number of sensors for (a) linear, (b) Franke function, and (c) nonpolynomial true field, with visible error bars depicting 1 std. error over 50 repeated simulations. For Padua sensor locations, we plot NMQA (open red circles) vs Lagrange polynomial interpolation (black dashed triangles). For a regular $d \times d$ grid, we plot NMQA (filled circles) vs radial basis functions (blue squares). Standard assignment approach is shown as a threshold using a fixed arrangement of nine sensor qubits and averaged over four nonunique orientations (gray solid line). Data shown for $m = 50$ measurements per sensor qubit, with $d = 2, 3, 4, 6, 9$ and $\kappa = 1, 2, 3, 4, 5, 10$.

measurement errors from quantum projection noise for each additional sensor increase the total error in the reconstruction. This increase in reconstruction error is illustrated by the gradual increase in mapping error with the sensor-qubit number for the black dashed triangles. The RBF interpolation strategy initially performs worse than Lagrange-Padua interpolation, but the two methods become approximately equivalent as more qubits are added, even though the field complexity remains fixed. In this case NMQA performs substantially worse than Lagrange-Padua interpolation for small qubit numbers, but its in-built noise filtering properties lead all methods to converge in the high m, κ regime.

Moving to other functional forms for f, our results for the Franke function in Fig. 2(b) show that the performance of the Lagrange-Padua method improves with increasing κ, but does not outperform other methods, particularly for small or moderate values of κ, where it is not even well distinguished from the standard assignment approach. RBF's superior performance in the high-data limit is somewhat unsurprising since Gaussian components of the Franke function are much stronger than the background linear drift, and these Gaussian spatial variations are naturally expressed in the functional basis of RBF. In this case, NMQA performs marginally better in the low-data regime but is outperformed by RBF in the large-measurement limit.

Finally, we observe strikingly different behavior when f is not approximated well by the natural functional basis of any interpolant. The final panel, Fig. 2(c), represents a challenging field reconstruction problem for all of our chosen methods. Interpolation using Lagrange polynomials on Padua grids vs RBF on regular grids shows that both interpolation methods perform similarly and provide little if any benefit relative to

the standard assignment approach. In this case we observe that NMQA substantially outperforms any other reconstruction strategy, illustrating the relevance of an adaptive measurement strategy when one cannot use an interpolant in some ideal functional basis relative to the true field.

A surprising feature of panels 2(a)–2(c) is that the error scaling behavior of NMQA appears to depend only on the total number of sensor qubits on the unit square, and not on the spatial arrangement of sensor qubits (Padua vs regular grid), unlike interpolation. The difference between NMQA and interpolation strategies can also be examined by varying the number of single-qubit measurements per sensor qubit, m. In Fig. 3, we plot mapping error vs m for two different field configurations (rows) and three different orders for the sensor grid (columns). The limit $m \rightarrow \infty$ implies that a true field is known perfectly on the locations of the sensor qubits for ideal interpolation, indicating that errors in functional evaluation are reduced from left to right along the x axis in each panel of Fig. 3.

In Fig. 3(a), $\kappa = n = 1$ is close to satisfying conditions for optimal interpolation on a linear field and the Lagrange-Padua method provides the lowest error reconstruction for all m. The opposite is true in 3(d)–(f), where NMQA significantly outperforms Lagrange-Padua on a nonpolynomial field. For the panels with $\kappa > 1$ the performance of Lagrange-Padua methods deteriorates with respect to NMQA due to quantum projection noise. In some cases [e.g., panels 3(b) and 3(c)], we see a crossover in performance between Lagrange-Padua interpolation and NMQA with m (as projection noise decreases with m). However, this behavior disappears in the more challenging field reconstruction settings of 3(e) and 3(f), where NMQA's benefits are more substantial. A comparison of open
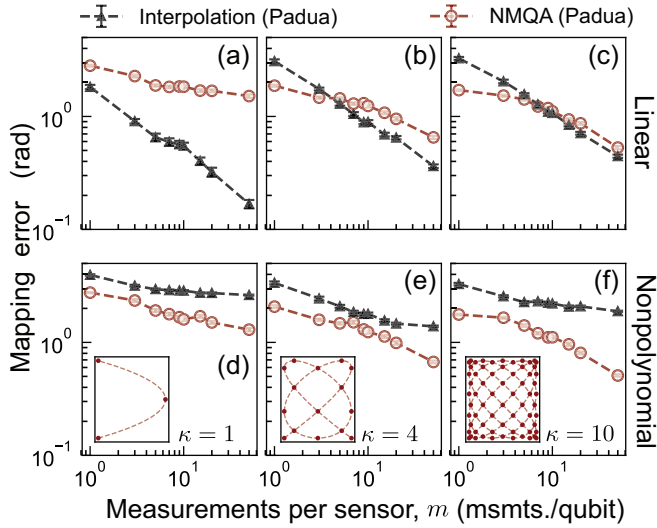
FIG. 3. Expected mapping error in radians (rad) vs measurements per sensor qubit m for linear field ($n = 1$) in top row (a)–(c) and nonpolynomial field in bottom row (d)–(f). Columns depict increasing order of the Padua grid from left to right $\kappa = 1, 4, 10$. In each panel, mapping error vs m number of measurements per sensor qubit is plotted for NMQA (open red circles) vs Lagrange polynomial interpolation (black dashed triangles) over 50 trials with visible error bars depicting 1 std. error. Padua sensor arrangements (red dots) and associated generating curve (red dashed) for each column is shown as bottom left insets in (d)–(f). For a linear field in (a), $\kappa = n = 1$ is optimal Lagrange-Padua interpolation for $m \to \infty$ and (b),(c) represent an unnecessary increase in the number of sensor qubits. In (d)–(f), a nonpolynomial true field permits no finite order polynomial representation.

circles in the top vs bottom row of panels shows that mapping error trajectories for NMQA do not change with the type of true field. This observation empirically confirms that, unlike interpolation, NMQA's efficacy depends on branching random processes [17], and hence its structure precludes choosing a functional basis for map reconstruction.

If *a priori* information about the true field is available, the results of Figs. 2 and 3 demonstrate that the appropriate configuration of Lagrange-Padua methods will yield the lowest error field reconstruction on unmeasured data qubits. If no such information is available, then NMQA satisfies our physical intuition that reconstruction performance improves with an increase in the number of sensor qubits, whereas this intuition is not always true for RBF or Lagrange interpolation if the interpolant is poorly specified with respect to the properties of the true field.

One anticipates that if κ is sufficiently high relative to the polynomial order of the true field, and functional evaluation errors approach zero for large m, then Lagrange-Padua interpolation will always reflect a low-error reconstruction relative to RBF and NMQA approaches. Since there is no finite polynomial representation of the test function in Fig. 2(c), we have not been able to numerically ascertain the value of κ that results in Lagrange-Padua interpolation outperforming NMQA.

IV. CONCLUSION

Under the spectator qubit paradigm, we investigated protocols to efficiently characterize spatial inhomogeneities in qubit calibration or performance, modeled as a classical scalar field in 2D. By collecting measurements on a dedicated sublattice of sensor qubits, we estimated the field values at the locations of proximal, unmeasured data qubits in a 2D multiqubit device, assuming a dephasing-type Hamiltonian. Drawing from optimal interpolation and statistical estimation theory, we used simulated data to compare the performance of reconstruction methods in the presence of different field configurations, studying the impact of the underlying sensor-qubit architecture of field characterization.

Our results showed that in most circumstances it is advantageous to arrange sensor qubits at the Padua points, an optimal point set for 2D interpolation. In circumstances where the field is well approximated by polynomial functions, Lagrange-Padua interpolation outperforms comparable interpolation strategies using regular sensor-qubit lattices, as well as adaptive measurement strategies. If, however, fields are not well approximated by polynomial functions, adaptive inference procedures such as NMQA perform best. Crucially, as we have shown here, the performance of NMQA shows limited sensitivity to the underlying sensor arrangement.

For complicated spatial variations, our results indicate that both finite-measurement effects and the presence of nonpolynomial field structure reduce the utility of optimal Lagrange-Padua or RBF interpolation strategies in a spectator-qubit paradigm. In some cases *a priori* information about spatial variation in a target field parameter may be available, but unless the interpolant can be appropriately specified, increasing the number of sensor qubits can increase overall field-reconstruction error. This observation runs counter to the physical intuition that providing more sensors in a field reconstruction problem should improve the quality of the reconstruction. By contrast, the noise-filtering properties of adaptive strategies such as NMQA enable the algorithm to reduce mapping error as the total amount of information increases (e.g., increasing sensor counts).

In this application NMQA also natively accommodates the potential to incorporate temporal drifts which are not compatible with interpolation strategies. The dynamical model in NMQA can be modified to include temporal dynamics that may either be specified *a priori* or learned through data. In the adaptive control [23] and probabilistic robotics literature [24], particularly in the context of spatiotemporal mapping applications for self-navigating vehicles [25], a time-varying environment is often sampled rapidly such that a formal dynamical model need not be specified and estimated state information is "forgotten" by the algorithm over some time scale. Contemporary inference proposals use hidden Markov autoregressive moving average models [26] or Gaussian processes [27] to introduce temporal correlations to an otherwise static pattern estimation problem [28–30]. A full spatiotemporal analysis of NMQA remains an exciting subject for future work.

A hybrid of both NMQA and Lagrange-Padua techniques yield a general protocol for spatial field characterization. For higher accuracy, one may first use NMQA for a

coarse-grained characterization of a target field, followed by a Lagrange-Padua method on local regions that are approximately polynomial (with interpolation order and number of measurements informed by the results of NMQA). For quasistatic linear or quadratic spatial variation, only first- or second-order polynomial interpolants are needed, in which case the Padua point set is an exact subset of a regular square grid, as explained in the Appendixes. Thus NMQA characterization and local refinements via Lagrange-Padua interpolation are practically viable even for regular grids. Alternatively, building 2D qubit arrays in Padua arrangements is equally attractive, as Padua-grid-configured hardware will not impede any NMQA-based characterization procedures. We look forward to greater exploration of how physical-layer quantum computer architectures may be impacted by the control and characterization strategies in use.

Access to the code base and data required to reproduce all figures is provided via Ref. [31].

APPENDIX A: GRID GEOMETRIES FOR NUMERICAL SIMULATION

In Fig. 4 we illustrate the grid arrangements that have been used throughout the analysis of the main text.

The top row of Fig. 4 shows four possible orientations for the standard local neighborhood assignment method that have been averaged over in all the data figures presented. Local neighborhood assignments cannot generally be defined uniquely in 2D. For a regular data-qubit grid defined on a unit square, we choose the first four nearest-neighbor data qubits for every sensor qubit, thereby maximizing the number of first nearest neighbors in regular lattice. This procedure can continue at the interior points of any regular data-qubit grid when the number of rows and columns are even. For an odd row or column, the first nearest neighbors are necessarily equivalent to a 1D case where two data qubits are allocated to each sensor. The intersection of odd rows and columns is a vertex about which no neighbors are defined. We consider this asymmetric case for a 5×5 data-qubit grid and we average over the four possible orientations when we present results of data simulations.

The middle and bottom rows depict the regular and Padua spatial grid arrangements that have been investigated in the scope of our empirical study. Observing the Padua point sets in the bottom row of Fig. 4, we see that the Padua locations for $\kappa = 1, 2$ are an exact subset of a regular square grid arrangement. This suggests that quasistatic spatial variations for
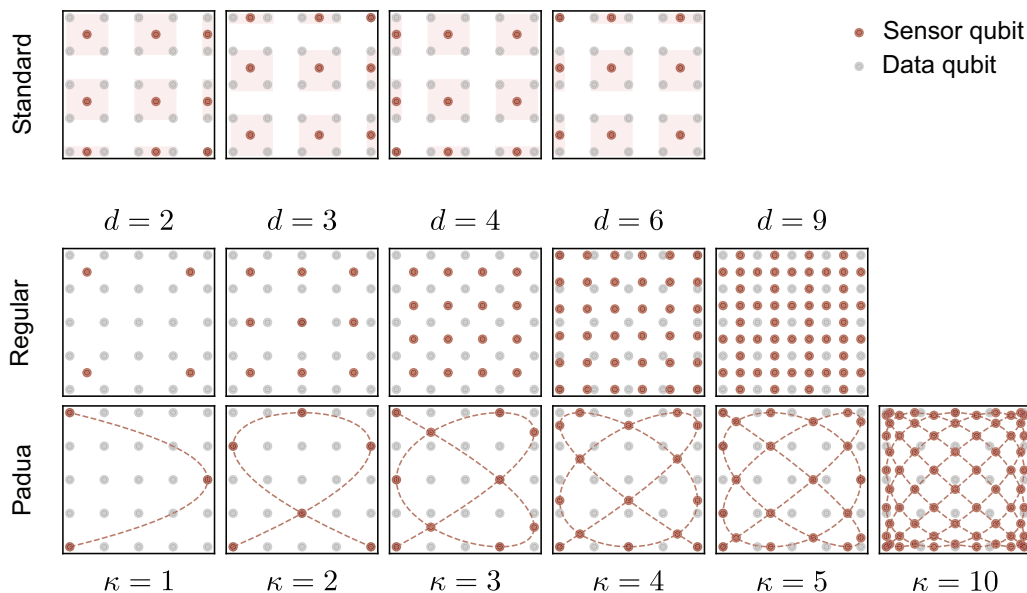

FIG. 4. Placement of sensor and data qubits on the unit square $[-1, 1]^2$ for standard local neighborhood value-assignment (top), regular (middle), and Padua (bottom) configurations. A 5×5 grid of data qubits (gray circles) is fixed in all panels, while sensor qubit (red circles) are varied. Top: four possible orientations of local neighborhood assignment of data qubits to a single sensor. Shaded regions depict fixed neighborhood groups. Middle: regular nested grid of $d \times d$ sensors, with $d = 2, 3, 4, 6$. For $d = 9$, sensor qubits that overlap exactly with data qubits are removed, leading to a total of 54 sensor qubits. Bottom: Padua locations for order $\kappa = 1, 2, 3, 4, 5, 10$ and associated generating curve (red dashed).

$\kappa \leqslant 2$ could be analyzed using Lagrange-Padua methods on specific types of regular grids.

APPENDIX B: IDEAL INTERPOLATION AND APPROXIMATION THEORY

The details for the formalism of ideal interpolation and approximation theory is the subject of this section. We restate the theoretical formalism following [32] and establish notation to present the relevant results of [13–16,33,34].

1. Least-squares criterion

Let $\mathcal{C}(\Omega)$ be the space of continuous functions of two variables defined on an open, connected, and bounded subset $\Omega \subset \mathbb{R}^2$ [32]. The geometry of Ω affects the interpolation problem and we focus on the case where $\Omega := [-1, 1]^2$ takes the form of a unit square. Our true function is denoted by $f \in \mathcal{C}(\Omega)$ and f is the subject of our approximation reconstruction over the region Ω.

We now wish to define two members in the space of bivariate polynomials. Let $\mathcal{P}_\kappa$ be the space of bivariate polynomials of degree at most κ with dimension N. Let the points $\chi := \{x_i\}_{i=1}^L$ denote the position of L known values of f, and define a set of associated positive weights as $\mathcal{W} := \{w_i\}_{i=1}^L \subset \mathbb{R}^+ \setminus \{0\}$. We define the best polynomial approximation to f as p^* and the minimum least-squares polynomial as p_L satisfying

$$\min_{p_L} \| f - p_L \|_2 = \min_{p_L} \sqrt{\sum_{i=1}^L w_i^2 |f(x_i) - p_L(x_i)|^2}. \quad \text{(B1)}$$

This minimum least-squares operator is often restated in approximation problems as the interpolant $\mathcal{L}$ that maps f to p_L, $\mathcal{L} : \mathcal{C}(\Omega) \to \mathcal{P}_\kappa$. The interpolant, $\mathcal{L}$, is thus a linear operator that depends on the point set χ, the weight set $\mathcal{W}$, and the approximation space $\mathcal{P}_\kappa$ parametrized by the value κ.

The least-squares approximating polynomial p_L can be found by writing $\mathcal{L}$ in an appropriate polynomial basis, and solving for p_L. We recap the specific inverse problem and the error properties of $\mathcal{L}$ in terms of χ and κ below.

Let $\{p_j\}_{j=1}^N$ be a polynomial basis for $\mathcal{P}_\kappa$ and $V_\chi := [p_j(x_i)]$ be the $L \times N$ Vandermonde matrix for this basis using the point set χ. The interpolant at some arbitrary location $x \in \Omega$ is written in terms of coefficients in this basis:

$$\mathcal{L}(f)(x) := \rho(x)^T c = \sum_{j=1}^N p_j(x) c_j, \quad x \in \Omega, \quad \text{(B2)}$$

$$c := [c_1 \quad c_2 \quad \ldots \quad c_N]^T, \quad \text{(B3)}$$

$$\rho := [p_1 \quad p_2 \quad \ldots \quad p_N]^T. \quad \text{(B4)}$$

Here, c represents N scalar coefficients; ρ represents an N-polynomial basis with each element corresponding to a basis polynomial p_j. The coefficients c are solutions to a linear system of equations $WV_\chi c = WF_\chi$, where

$$W := \text{diag}(w_1 \ldots w_L), \quad \text{(B5)}$$

$$F_\chi := [f(x_1) \quad f(x_2) \quad \ldots f(x_L)], \quad \{x_i\}_{i=1}^L \in \chi. \quad \text{(B6)}$$

The coefficients are thus determined via optimization or using the pseudoinverse $c = (WV_\chi)^\dagger WF_\chi$, yielding

$$\mathcal{L}(f)(x) := \rho(x)^T (WV_\chi)^\dagger WF_\chi, \quad x \in \Omega. \quad \text{(B7)}$$

Here, the matrix W is invertible by construction. The pseudoinverse governs error properties of the interpolant $\mathcal{L}$.

Following [32], the statement that the solution c is unique is equivalent to the condition that V_χ has full rank. If additionally $L = N$, then the point set χ is called unisolvent for which V_χ is invertible and the interpolant simplifies to

$$\mathcal{L}(f)(x) := \rho(x)^T V_\chi^{-1} F_\chi, \quad x \in \Omega. \quad \text{(B8)}$$

2. Conditioning of the interpolant

Using notation in the previous section, we now examine least-squares solutions and add perturbations to the linear system of the previous section. In practical application of interpolation theory, these perturbations arise from inaccuracies in our choice of basis V_χ or the point set χ that lead to inaccuracies in the solution c. For the purposes of this section, we present the concept of a condition number $\eta(V)$ for a choice of a basis for the matrix V. The condition number is unity for an optimal basis, which means that any errors in the matrix V are not magnified as errors in the solution c. These errors are to be contrasted with the effect of measurement noise or errors in the data set, which will be presented in future subsections, and thus F_χ is noiseless in the discussion below.

Let the following equations represent a true linear system of equations and a perturbed linear system, denoted via ˜ as

$$WV_\chi c = WF_\chi, \quad \text{(B9)}$$

$$W\tilde{V}_\chi \tilde{c} = WF_\chi. \quad \text{(B10)}$$

Next, we impose the ∞ or the uniform norm which satisfies consistency when applied to matrices and vectors such that $\|Ax\| \leqslant \|A\|\|x\|$, where A and x represent arbitrary matrices and vectors. Let E be an error matrix such that $\tilde{V}_\chi = V_\chi + E$

$$WF_\chi = W(V_\chi + E)\tilde{c} = WV_\chi c \quad \text{(B11)}$$

$$\Rightarrow (WV_\chi + WE)\tilde{c} = WV_\chi c \quad \text{(B12)}$$

$$\Rightarrow [I + (WV_\chi)^\dagger WE]\tilde{c} = c \quad \text{(B13)}$$

$$\Rightarrow \tilde{c} - c = (WV_\chi)^\dagger WE)\tilde{c} \quad \text{(B14)}$$

$$\Rightarrow \|\tilde{c} - c\| \leqslant \|(WV_\chi)^\dagger WE\|\|\tilde{c}\|. \quad \text{(B15)}$$

In the above simplification, we assume $(WV_\chi)^\dagger$ is a left pseudoinverse and we apply consistency of the ∞ norm. If the quantity $\|(WV_\chi)^\dagger WE\| < 1$, then the error in the estimated interpolation coefficients $\tilde{c}$ relative to the true (unknown) coefficients c can be written as the following inequality [35]:

$$\frac{\|\tilde{c} - c\|}{\|c\|} \leqslant \frac{\|(WV_\chi)^\dagger WE\|}{(1 - \|(WV_\chi)^\dagger WE\|)}. \quad \text{(B16)}$$

To interpret this inequality, we explicitly introduce the condition number by expanding and rewriting the term

$\|(WV_\chi)^\dagger WE)\|$ in terms of the condition number $\eta(V)$:

$$\left\|(WV_\chi)^\dagger WE)\right\| \leqslant \left\|(WV_\chi)^\dagger\right\| \|WE\| \tag{B17}$$

$$= \left\|(WV_\chi)^\dagger\right\| \|W\tilde{V}_\chi - WV_\chi\| \tag{B18}$$

$$= \eta(V)\frac{\|W\tilde{V}_\chi - WV_\chi\|}{\|WV_\chi\|} \tag{B19}$$

$$\eta(V) := \left\|(WV_\chi)^\dagger\right\| \|WV_\chi\|. \tag{B20}$$

In this derivation, we apply the consistency of the uniform norm and we substitute the expression for E. The norm of the matrix WV_χ is added as a dummy variable, yielding the final expression for the condition number. When $\|(WV_\chi)^\dagger WE)\| < 1$, we can write the following approximate inequality that links the condition number of V_χ as the magnification of true errors when any errors in the matrix V_χ are passed on to the solution for c:

$$\frac{\|\tilde{c} - c\|}{\|c\|} \leqslant \eta(V)\frac{\|W\tilde{V}_\chi - WV_\chi\|}{\|WV_\chi\|}. \tag{B21}$$

It is well known that the optimal condition number is one and, in general, $\eta(V) \geqslant 1$ [35].

The effect of the condition number on the ∞ norm of the interpolant can be seen by rearranging terms to obtain $\|\mathcal{L}(f)\| \leqslant \eta(V)\|\rho(x)^T\| \|c\|$. Note that the condition number does not depend on f but only the structure of the linear system which we are trying to solve. Hence, for a general point set χ and choice of polynomial basis for $\mathcal{P}_\kappa$, $\eta(V)$ can be large.

Under an optimal choice of basis, however, the condition number satisfies $\eta(V) = 1$, i.e., errors are not magnified in the inversion process [32]. This optimal basis for the least-squares problem is in fact an orthonormal polynomial basis with respect to a discrete inner product over χ [32]:

$$\langle p_i, p_j \rangle := \sum_{k=1}^{L} w_k^2 p_i(x_k)p_j(x_k) \tag{B22}$$

$$= \delta_{i,j}, \quad \{x_k\}_{k=1}^{L} \in \chi, \ \{p_i\}_{i=1}^{N}. \tag{B23}$$

The geometry of Ω influences whether an optimal basis can be found for a given application.

3. Lebesgue constant

The ∞ norm of the interpolant $\|\mathcal{L}(f)\|$ is typically used in approximation error analysis—in particular, to answer the question "how good is $\mathcal{L}(f)$ as an approximation to f?" To address this question, one seeks the so-called Lebesgue constant, $\Lambda_\mathcal{L}$, to provide the lowest upper bound on the norm of the interpolant:

$$\Lambda_\mathcal{L} := \min\{\lambda \geqslant 0 | \|\mathcal{L}(f)\| \leqslant \lambda\|f\|, \quad \forall f \in \mathcal{C}(\Omega)\}. \tag{B24}$$

Here, the constant $\Lambda_\mathcal{L}$ is independent of the form of the function f or the construction of $\mathcal{L}(f)$. Hence the behavior of $\Lambda_\mathcal{L}$ enables a definition of optimal across different interpolation strategies or different geometries.

In addition to providing a bound on the ∞ norm of the interpolant, we can also use $\Lambda_\mathcal{L}$ to provide an upper bound on approximation error. To do this, we consider three points

in the space of continuous polynomials and apply the triangle inequality [32]. Since $\mathcal{P}_\kappa \subset \mathcal{C}(\Omega)$, we may consider $f, \mathcal{L}(f)$ and the best approximating polynomial p^* as points in the space of continuous functions on Ω. One applies the triangle inequality to obtain

$$\| f - \mathcal{L}(f) \| \leqslant \| f - p^* \| + \| p^* - \mathcal{L}(f) \|, \quad \forall f \in \mathcal{C}(\Omega). \tag{B25}$$

Here, the best approximating polynomial is optimal with respect to the ∞ norm $\|f - p^*\|$, and thus it should be distinguished from the least-squares criterion used to construct the interpolant $\mathcal{L}(f)$.

Using $\mathcal{L}(p^*) := p^*$ and the linearity of the interpolant, one substitutes

$$\| p^* - \mathcal{L}(f) \| = \| \mathcal{L}(p^* - f) \| \tag{B26}$$

$$\leqslant \Lambda_\mathcal{L} \| p^* - f \| \tag{B27}$$

$$\Rightarrow \|f - \mathcal{L}(f)\| \leqslant (1 + \Lambda_\mathcal{L})\|f - p^*\|. \tag{B28}$$

The last step combines the two inequality relations and enables the interpretation of the Lebesgue constant as a measure of how much worse an interpolant performs with respect to some best approximating polynomial p^*.

APPENDIX C: SOURCES OF ERROR

1. Measurement errors in true f

In this section, we analyze the effect of errors in the functional values at the point set χ. Let $\tilde{f}$ denote the perturbed function for the true f, with the error in functional values expressed as

$$\epsilon(x) := f(x) - \tilde{f}(x), \quad f, \tilde{f} \in \mathcal{C}(\Omega); x \in \chi. \tag{C1}$$

The error in the interpolant is thus linear by the linearity of the operator $\mathcal{L}$:

$$\mathcal{L}(f) - \mathcal{L}(\tilde{f}) := \mathcal{L}(f - \tilde{f}) \tag{C2}$$

$$= \mathcal{L}(\epsilon). \tag{C3}$$

Letting F_χ and $\tilde{F}_\chi$ denote the corresponding vectors where each element is computed using f and $\tilde{f}$, respectively, we obtain an expression for the error in the interpolant:

$$\mathcal{L}(\epsilon)(x) := \rho(x)^T (WV_\chi)^\dagger W(F_\chi - \tilde{F}_\chi), \quad x \in \Omega. \tag{C4}$$

If we assume further that $\epsilon \in \mathcal{C}(\Omega)$, we can use the definition of the Lebesgue constant:

$$\|\mathcal{L}(\epsilon)\| \leqslant \Lambda_\mathcal{L}\|f - \tilde{f}\|. \tag{C5}$$

.

Under the specific condition that $\|\mathcal{L}(f)\| \geqslant \|f\|$, the inequality above can be cast to establish $\|\mathcal{L}(f)\|$ as the condition number for passing on errors in functional values to the interpolant:

$$\frac{\|\mathcal{L}(\epsilon)\|}{\|\mathcal{L}(f)\|} \leqslant \Lambda_\mathcal{L}\frac{\|f - \tilde{f}\|}{\|f\|}. \tag{C6}$$

In 1D, this condition is met for the Lagrange polynomials, leading to the straightforward interpretation that interpolation strategies with minimal $\Lambda_\mathcal{L}$ reduce sensitivity of the resulting

interpolant to errors in functional values (see [33] for 1D case). The condition $\|\mathcal{L}(f)\| \geqslant \|f\|$ is generally not true for an arbitrary polynomial basis for an interpolation strategy in 2D. For the specific case where χ represents the Padua point set on a unit square, one can derive the optimality relations discussed in subsequent sections.

2. Perturbed point set χ

In this section, we assess the impact of a perturbed point set χ. For a chosen χ and polynomial basis for $\mathcal{P}_\kappa$, finite size effects mean that accessing exact $x \in \chi$ during practical applications is impossible.

There is very limited research on this subject (see [36] for the case of 1D continuous function on an equispaced grid with a fixed perturbation) and as yet no analysis for bivariate interpolation problems on specific geometries (e.g., the unit square).

In lieu of a formal derivation of error bounds, we consider the case that each point x in the selected point set χ is weakly perturbed by a fixed displacement $\epsilon \to 0$:

$$\epsilon := \tilde{x} - x, \quad \forall x \in \chi. \tag{C7}$$

The introduction of these errors now results in two nonlinear perturbations: first, in the elements of $V_\chi \to V_{\chi,\epsilon}$ and, similarly, $F_\chi \to F_{\chi,\epsilon}$, where the subscript $_{\chi,\epsilon}$ denotes that the matrix or vector elements are being computed using the perturbed points $\tilde{x}$.

For weak noise, the size of the perturbations depend on the derivatives of the function f and the chosen polynomial basis $\{p_j\}$ for the space $\mathcal{P}_\kappa$.

A first-order Taylor expansion recasts the effect of these perturbations on the functional values $F_{\chi,\epsilon}$ as approximately linear:

$$f(\tilde{x}) := f(x) + \epsilon f'(x) + O(\epsilon^2) \tag{C8}$$

$$\Rightarrow F_{\chi,\epsilon} := F_\chi + \epsilon F' + O(\epsilon^2) \tag{C9}$$

$$F' := [f'(x_1) \quad \dots \quad f'(x_L)]. \tag{C10}$$

Thus, to first order, $F_{\chi,\epsilon}$ will manifest as measurement errors of the previous section. Higher orders may be considered depending on the specific form of f. The strength of the errors will depend on the derivatives of f.

Similarly, first-order expansions of the basis polynomials lead to an expression for the perturbed Vandermonde matrix:

$$p_j(\tilde{x}_i) := p_j(x_i) + \epsilon p'_j(x_i) + O(\epsilon^2) \tag{C11}$$

$$\Rightarrow V_{\chi,\epsilon} := V_\chi + \epsilon V' + O(\epsilon^2) \tag{C12}$$

$$V' := [p'_j(x_i)] \tag{C13}$$

$$j = 1, \dots, N \tag{C14}$$

$$i = 1, \dots, L. \tag{C15}$$

We now observe that the quantity V' will have a first column consisting of all zeros and the remaining elements form the basis for the polynomial space $\mathcal{P}_{\kappa-1}$. Hence V' is singular, but a pseudoinverse for the quantity WV' exists. We substitute these new matrices into the interpolant formula to get

$$\mathcal{L}(\epsilon)(x) := \rho(x)^T (WV_\chi + \epsilon WV')^\dagger W(F_\chi + \epsilon F'), \quad x \in \Omega. \tag{C16}$$

We use the properties of the Moore-Penrose pseudoinverse for any pair of $m \times n$ matrices A and B [37],

$$(A + B)^\dagger = \frac{1}{2}\begin{bmatrix} I_n & I_n \end{bmatrix} \begin{bmatrix} A & B \\ B & A \end{bmatrix}^\dagger \begin{bmatrix} I_m \\ I_m \end{bmatrix}, \tag{C17}$$

to rewrite the pseudoinverse of the first- and second-order $L \times N$ Vandermonde matrices:

$$\mathcal{V}(\epsilon) := (WV_\chi + \epsilon WV')^\dagger \tag{C18}$$

$$= \frac{1}{2}\begin{bmatrix} I_N & I_N \end{bmatrix} \begin{bmatrix} WV_\chi & \epsilon WV' \\ \epsilon WV' & WV_\chi \end{bmatrix}^\dagger \begin{bmatrix} I_L \\ I_L \end{bmatrix} \tag{C19}$$

$$= \frac{1}{2\epsilon}\begin{bmatrix} I_N & I_N \end{bmatrix} \begin{bmatrix} WV_\chi/\epsilon & WV' \\ WV' & WV_\chi/\epsilon \end{bmatrix}^\dagger \begin{bmatrix} I_L \\ I_L \end{bmatrix}, \quad \epsilon \neq 0, \tag{C20}$$

where the last line is obtained by using the basic property that the pseudoinverse of any nonzero scalar multiple of matrix A is the reciprocal multiple of the pseudoinverse $A^\dagger$ satisfying $(\epsilon A)^\dagger = \epsilon^{-1} A^\dagger$, for some $\epsilon \neq 0$.

Under the weak perturbation approximation, and to first order in ϵ, we see that the combined effect of a perturbed χ manifests as both errors in $\mathcal{V}(\epsilon)$ and in functional evaluation:

$$\mathcal{L}(\epsilon)(x) = \frac{1}{2}\rho(x)^T \begin{bmatrix} I_N & I_N \end{bmatrix} \begin{bmatrix} WV_\chi/\epsilon & WV' \\ WV' & WV_\chi/\epsilon \end{bmatrix}^\dagger \begin{bmatrix} I_L \\ I_L \end{bmatrix}$$
$$\times W\left(\frac{F_\chi}{\epsilon} + F'\right),$$
$$\epsilon \neq 0. \tag{C21}$$

In this form, it appears that the total error will be mediated by the condition number $\eta(\mathcal{V}(\epsilon))$ and additional magnification of order $\frac{1}{|\epsilon|}$ in functional evaluation in the weak error limit $|\epsilon| \to 0$. For f approximately slowly varying or constant, we can ignore errors in functional evaluation caused by the perturbed grid and set $F' \approx 0$, thereby noticing that an error ϵ inside the pseudoinverse term and the functional evaluation term do not easily cancel out in general and only cancel if the off-diagonal first-order terms additionally satisfy $V' \approx 0$. It remains an open question to see if a perturbed grid can be recast as functional evaluation errors under some other considerations.

APPENDIX D: BIVARIATE POLYNOMIAL INTERPOLATION AT THE PADUA POINTS

1. Optimality on the unit square

For the unit square, $\Omega := [-1, 1]^2$, it was discovered that the product Chebshev polynomials evaluated at the so-called Padua point set χ_p and with the correct theoretically derived weights W_p form an optimal basis with respect to the discrete inner product. Here, the subscript $_p$ denotes Padua-based interpolation strategies. Further, the interpolation problem presented in earlier sections turns out to be unisolvent, enabling a unique least-squares solution for the interpolants c_p and the matrix $W_p V_{\chi_p}$ turns out to have a unity condition number [32].

For the set of 2D problems mappable to the unit square, bivariate Lagrange interpolation at the Padua points also yields

the slowest-growing error bound for approximation error as $\Lambda_{\mathcal{L}_p} \sim O(\log^2(\kappa))$, where κ denotes the order of the Padua points to interpolate f at most degree κ [13–15]. In particular, for some constant $a(f, k)$ that depends on $f \in \mathcal{C}^k(\Omega)$ and its k continuous derivatives [16],

$$\|f - \mathcal{L}(f)\| \leqslant (1 + \Lambda_{\mathcal{L}})\|f - p^*\| \leqslant a(f, k)\frac{\log^2(\kappa)}{\kappa^k}. \quad \text{(D1)}$$

Here, κ represents both the space of polynomials of degree at most κ as well as the order of the Padua point set; k denotes the number of continuous derivatives and a is a constant that depends on both the function f and k.

2. Lagrange-Padua interpolation

Padua points can be generated in three equivalent ways: (a) via the use of a generating curve approach [13], the ideal theory approach [14], and through the merger of two Chebychev-like grids [15]. There are four families of Padua points and we focus on the first family for the equations below.

Let $x \in \chi$ be a point in the set of Padua points χ of order κ over the unit square $\Omega := [-1, 1]^2$. The number of Padua points depends on the order, κ, as

$$|\chi_\kappa| := \frac{(\kappa + 2)(\kappa + 1)}{2}, \quad \kappa > 0. \quad \text{(D2)}$$

The geometric interpretation of Padua points can be obtained by envisioning them as equally spaced points along a generating curve, $\gamma_\kappa(t)$, on the unit square, Ω. The intersection of this generating curve with itself, the edges of the square, or its vertices yields the Padua point set. The generating curve can be defined as

$$\gamma_\kappa(t) := (-\cos[(\kappa + 1)t], -\cos(\kappa t)). \quad \text{(D3)}$$

On this curve, the Padua points are equispaced along t and indexed by j, m as

$$t_{(j,m)} := \frac{j\kappa + m(\kappa + 1)}{\kappa(\kappa + 1)}\pi, \quad j, m \geqslant 0, \quad j + m \leqslant \kappa. \quad \text{(D4)}$$

The set of Padua points are classified as interior, boundary, or vertex points. Two vertex points occur at $(1, 1)$ and $((-1)^\kappa, (-1)^{\kappa+1})$ and edge points occur on the boundary of the square. The curve of Eq. (D3) is consistent with [15].

The so-called cubature weight w_x of the point x depends on its classification:

$$w_x := \frac{1}{\kappa(\kappa + 1)}\begin{cases} 1/2, & x \text{ vertex,} \\ 1, & x \text{ boundary,} \\ 2, & x \text{ interior.} \end{cases} \quad \text{(D5)}$$

The cubature weights above agree with [15].

The formulas above enable interpolation for any true f using Lagrange polynomial interpolation of degree at most κ. The interpolation formula is written as [13,14]

$$\mathcal{L}(f)(x') := \sum_{x \in \mathbb{P}_\kappa} f(x)w_x(K_\kappa(x, x') - T_\kappa(x[0])T_\kappa(x'[0])) \quad \text{(D6)}$$

$$= \sum_{x \in \mathbb{P}_\kappa} f(x)l(x, x'), \quad \text{(D7)}$$

$$l(x, x') := \frac{K^*(x, x')}{K^*(x, x)}, \quad \text{(D8)}$$

$$K^*(x, y) := K_\kappa(x, y) - T_\kappa(x[0])T_\kappa(y[0]), \quad \text{(D9)}$$

$$K_\kappa(x, y) := \sum_j^\kappa \sum_i^j T_i(x[0])T_{j-i}(x[1])T_i(y[0])T_{j-i}(y[1]), \quad \text{(D10)}$$

$$w_x := \frac{1}{K^*(x, x)}, \quad \text{(D11)}$$

where $(x[0], x[1])$ are the coordinates of x in 2D, $T_\kappa(\cdot)$ are the Chebychev polynomials of order κ, and $K_\kappa(x, y)$ is a reproducing kernel for the space of bivariate polynomials on the unit square with degree at most κ [14,15]. The Lagrange basis polynomials form an orthonormal basis over the Padua points, having the property $l(x_1, x_2) = 1 \iff x_1 = x_2$ and zero otherwise, for any two Padua points $x_1, x_2 \in \chi_\kappa$.

We supplement the geometric picture of Padua points with an alternative formulation that enables rapid calculation. In this alternative picture, the Padua points are a subset of a grid of Chebyshev-Gauss-Lobatto points, $\chi_\kappa \subset C_{\kappa+1} \times C_{\kappa+2}$. The interpolant can be written in terms of a matrix of coefficients, $\mathbb{C}_0(\cdot)$, and a rectangular Chebyshev matrix $\mathbb{T}(\cdot)$:

$$\mathcal{L}(f)(\mathbf{X}) := ((\mathbb{T}(X_1))^t \mathbb{C}_0(f)\mathbb{T}(X_2))^t, \quad \text{(D12)}$$

$$\mathbb{T}(S) := \begin{bmatrix} \hat{T}_0(s_1) & \dots & \hat{T}_0(s_m) \\ \vdots & \dots & \vdots \\ \hat{T}_\kappa(s_1) & \dots & \hat{T}_\kappa(s_m) \end{bmatrix}, \quad \text{(D13)}$$

$$S = [s_1, \dots, s_m], \quad \hat{T}_y(\cdot) := \sqrt{2}T_y(\cdot). \quad \text{(D14)}$$

Here, $\mathbf{X} := X_1 \times X_2$ is a discrete Cartesian grid; X_i is a vector of the ith coordinate of all test points for the interpolation of the function f. The notation ^{t} denotes a matrix transpose. The Chebyshev matrix $\mathbb{T}(X_i)$ has dimensions $\kappa \times \dims(X_i)$ and has elements given by the scaled Chebyshev polynomials $\sqrt{2}T_y(\cdot)$ of order y.

The matrix of coefficients, $\mathbb{C}_0(f)$, is computed as essentially the left upper triangular component of a $(\kappa + 1) \times (\kappa + 1)$ square matrix $\mathbb{C}(f)$:

$$\mathbb{C}(f) := \mathbb{T}(C_{\kappa+1})\mathbb{G}(f)(\mathbb{T}(C_{\kappa+2}))^t. \quad \text{(D15)}$$

There is one modification: the last element of the first column of $\mathbb{C}_0(f)$ is multiplied by a factor of one-half. The construction of the remaining matrices will be discussed below.

The rectangular Chebyshev matrices are now defined by a vectorized set of Chebyshev-Gauss-Lobatto points of order $\kappa + 1$:

$$C_{\kappa+1} := \left\{ z_j^\kappa = -\cos((j - 1)\pi/\kappa), \quad j = 1, \dots, \kappa + 1 \right\}. \quad \text{(D16)}$$

In the above, the negative sign is required so that the Padua points, generating curve, and the Chebyshev grids yield the same point set.

Lastly, the $(\kappa + 1) \times (\kappa + 2)$ matrix $\mathbb{G}(f)$ incorporates the effect of the values of f evaluated on the Padua point set, as well as the Padua curbature weights. Entries of this matrix are nonzero only if the index of the matrix element coincides with

a Padua point:

$$\mathbb{G}(f) := (g_{r,s}), \tag{D17}$$

$$= \begin{cases} w_x f(x), & \text{if } x = (z_r^{\kappa}, z_s^{\kappa+1}) \in \chi_{\kappa}, \\ 0, & \text{if } (z_r^{\kappa}, z_s^{\kappa+1}) \in C_{\kappa+1} \times C_{\kappa+2} \setminus \chi_{\kappa}. \end{cases} \tag{D18}$$

The entries of $\mathbb{G}(f)$ which coincide with the Padua points can be quickly discovered by selecting only every other point in flattened "mesh grid" of $(\kappa + 1) \times (\kappa + 2)$ entries. The flattened mask is subsequently reshaped into a 2D matrix for odd values of κ as $(\kappa + 1) \times (\kappa + 2)$. For even values of κ, the mask is reshaped to $(\kappa + 2) \times (\kappa + 1)$, followed by a matrix transpose. If this mask is applied to the mesh grid of the Chebyshev-Gauss-Lobatto points $C_{\kappa+1} \times C_{\kappa+2}$, the Padua points of the generating curve approach in the previous section are recovered. Computationally, it is easier to construct $\mathbb{G}(f)$ by generating and merging the two Chebyshev grids, $C_{\kappa+1}, C_{\kappa+2}$, to obtain the Padua points.

The connection with the geometric interpretation is easier to see if we rewrite the Padua points with slightly modified index notation [14]:

$$x[0] := \cos \frac{k\pi}{\kappa}, \quad 0 \leqslant k \leqslant \kappa, \tag{D19}$$

$$x[1] := \begin{cases} \cos \frac{(2j-1)\pi}{(\kappa+1)}, & k \text{ even}, \\ \cos \frac{(2j-2)\pi}{(\kappa+1)}, & k \text{ odd}, \end{cases} \quad j = 1, \dots \lceil \kappa/2 \rceil + 1. \tag{D20}$$

The formulas above result in duplicate Padua points that can be removed by inspection; direct comparison with the definition of the Chebyshev-Gauss-Lobatto points confirms that $\chi_{\kappa} \subset C_{\kappa+1} \times C_{\kappa+2}$.

Two additional modifications to our Python codebase are required: first, the $\mathcal{L}(f)(\mathbf{X})$ matrix is flipped from left to right (corresponding to the use of the left upper triangular matrix of coefficients). Secondly, $\mathcal{L}(f)(\mathbf{X})$ is globally divided by a factor of four to compensate for the scaling factors in the rectangular Chebyshev matrix $\mathbb{T}(\cdot)$. Both of these modifications yield the final result in the correct orientation consistent with [15].

[1] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[2] J. Kelly, P. O'Malley, M. Neeley, H. Neven, and J. M. Martinis, Physical qubit calibration on a directed acyclic graph, arXiv:1803.03226.

[3] A. D. Tranter, H. J. Slatyer, M. R. Hush, A. C. Leung, J. L. Everett, K. V. Paul, P. Vernaz-Gris, P. K. Lam, B. C. Buchler, and G. T. Campbell, Multiparameter optimisation of a magneto-optical trap using deep learning, Nat. Commun. **9**, 4360 (2018).

[4] D. T. Lennon, H. Moon, L. C. Camenzind, L. Yu, D. M. Zumbühl, G. A. D. Briggs, M. A. Osborne, E. A. Laird, and N. Ares, Efficiently measuring a quantum device using machine learning, npj Quantum Inf. **5**, 79 (2019).

[5] P. B. Wigley, P. J. Everitt, A. van den Hengel, J. W. Bastian, M. A. Sooriyabandara, G. D. McDonald, K. S. Hardman, C. D. Quinlivan, P. Manju, C. C. N. Kuhn *et al.*, Fast machine-learning online optimization of ultra-cold-atom experiments, Sci. Rep. **6**, 25890 (2016).

[6] S. K. Tolpygo, V. Bolkhovsky, T. J. Weir, L. M. Johnson, M. A. Gouker, and W. D. Oliver, Fabrication process and properties of fully-planarized deep-submicron Nb/Al- AlO$_x$/Nb Josephson junctions for VLSI circuits, IEEE Trans. Appl. Supercond. **25**, 1 (2015).

[7] D. Koch, B. Martin, S. Patel, L. Wessing, and P. M. Alsing, Demonstrating NISQ era challenges in algorithm design on IBM's 20 qubit quantum computer, AIP Advances **10**, 095101 (2020).

[8] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, Fernando G. S. L. Brandao, D. A. Buell *et al.*, Quantum supremacy using a programmable superconducting processor, Nature (London) **574**, 505 (2019).

[9] S. Majumder, L. A. de Castro, and K. R. Brown, Real-time calibration with spectator qubits, npj Quantum Inf. **6**, 19 (2020).

[10] S. Shankar, M. Hatridge, Z. Leghtas, K. M. Sliwa, A. Narla, U. Vool, S. M. Girvin, L. Frunzio, M. Mirrahimi, and M. H

Devoret, Autonomously stabilized entanglement between two superconducting quantum bits, Nature (London) **504**, 419 (2013).

[11] O. Landon-Cardinal, L. C. G. Govia, and A. A. Clerk, Quantitative Tomography for Continuous Variable Quantum Systems, Phys. Rev. Lett. **120**, 090501 (2018).

[12] M. Caliari, S. De Marchi, and M. Vianello, Bivariate polynomial interpolation on the square at new nodal sets, Appl. Math. Comput. **165**, 261 (2005).

[13] L. Bos, M. Caliari, S. De Marchi, M. Vianello, and Y. Xu, Bivariate Lagrange interpolation at the Padua points: the generating curve approach, J. Approx. Theory **143**, 15 (2006).

[14] L. Bos, S. De Marchi, M. Vianello, and Y. Xu, Bivariate Lagrange interpolation at the Padua points: The ideal theory approach, Numer. Math. **108**, 43 (2007).

[15] M. Caliari, S. De Marchi, A. Sommariva, and M. Vianello, Padua2DM: Fast interpolation and cubature at the Padua points in Matlab/Octave, Numer. Algorithms **56**, 45 (2011).

[16] M. Caliari, S. De Marchi, and M. Vianello, Bivariate Lagrange interpolation at the Padua points: Computational aspects, J. Comput. Appl. Math. **221**, 284 (2008).

[17] R. S. Gupta and M. J. Biercuk, Convergence analysis for autonomous adaptive learning applied to quantum architectures, arXiv:1911.05752 [quant-ph].

[18] R. S. Gupta, C. L. Edmunds, A. R. Milne, C. Hempel, and M. J. Biercuk, Adaptive characterization of spatially inhomogeneous fields and errors in qubit registers, npj Quantum Inf. **6**, 53 (2020).

[19] L. M. K. Vandersypen and I. L. Chuang, NMR techniques for quantum control and computation, Rev. Mod. Phys. **76**, 1037 (2005).

[20] Z. Majdisova and V. Skala, Radial basis function approximations: comparison and applications, Appl. Math. Model. **51**, 728 (2017).

[21] M. L. Stein, *Interpolation of Spatial Data: Some Theory for Kriging* (Springer Science & Business Media, New York, 2012).

[22] R. Franke, A critical comparison of some methods for interpolation of scattered data, Tech. Rep. ADA081688 (Naval Postgraduate School, Monterey CA, 1979), https://apps.dtic.mil/sti/citations/ADA081688.

[23] I. D. Landau, R. Lozano, M. M'Saad, and A. Karimi, *Adaptive Control: Algorithms, Analysis and Applications* (Springer Science & Business Media, New York, 2011).

[24] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics* (MIT Press, Cambridge, MA, 2000), Vol. 1.

[25] B. Stanley (private communication).

[26] S. Michalek, M. Wagner, and J. Timmer, A new approximate likelihood estimator for ARMA-filtered hidden Markov models, IEEE Trans. Signal Process. **48**, 1537 (2000).

[27] C. K. I. Williams and C. Edward Rasmussen, *Gaussian Processes for Machine Learning*, Vol. 2 (MIT Press, Cambridge, MA, 2006).

[28] J. Luttinen and A. Ilin, Efficient Gaussian process inference for short-scale spatio-temporal modeling, in *Artificial Intelligence and Statistics* (Proceedings of Machine Learning Research, La Palma, Canary Islands, 2012), pp. 741–750.

[29] J. Hartikainen, J. Riihimäki, and S. Särkkä, Sparse spatio-temporal Gaussian processes with general likelihoods, in *Inter-national Conference on Artificial Neural Networks* (Springer, Berlin, 2011), pp. 193–200.

[30] K. Li and Y. Fu, ARMA-HMM: A new approach for early recognition of human activity, in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)* (IEEE, New York, 2012), pp. 1779–1782.

[31] R. S Gupta, *NMQA* (2020), GitHub repository, version 0.0.1, http://github.com/qcl-sydney/nmqa.

[32] M. Van Barel and M. Humet, Good point sets and corresponding weights for bivariate discrete least squares approximation, Dolomites Res. Notes Approx. **8**, 37 (2015).

[33] B. A. Ibrahimoglu, Lebesgue functions and Lebesgue constants in polynomial interpolation, J. Inequal. Appl. **2016**, 93 (2016).

[34] Y. Kolomoitsev, T. Lomako, and J. Prestin, On Lp-error of bivariate polynomial interpolation on the square, J. Approx. Theory **229**, 13 (2018).

[35] G. W. Stewart, *Afternotes on Numerical Analysis* (SIAM, Philadelphia, PA, 1996), Vol. 46.

[36] A. P. Austin and L. N. Trefethen, Trigonometric interpolation and quadrature in perturbed points, SIAM J. Numer. Anal. **55**, 2113 (2017).

[37] Y. Tian, The Moore-Penrose inverse for sums of matrices under rank additivity conditions, Lin. Multilin. Alg. **53**, 45 (2005).