SIAM/ASA J. Uncertainty Quantification
Vol. 12, No. 2, pp. 213–241

Subsampling of Parametric Models with Bifidelity Boosting*

Nuojin Cheng†, Osman Asif Malik‡, Yiming Xu§,
Stephen Becker†, Alireza Doostan¶, and Akil Narayan‖

Abstract. Least squares regression is a ubiquitous tool for building emulators (a.k.a. surrogate models) of problems across science and engineering for purposes such as design space exploration and uncertainty quantification. When the regression data are generated using an experimental design process (e.g., a quadrature grid) involving computationally expensive models, or when the data size is large, sketching techniques have shown promise at reducing the cost of the construction of the regression model while ensuring accuracy comparable to that of the full data. However, random sketching strategies, such as those based on leverage scores, lead to regression errors that are random and may exhibit large variability. To mitigate this issue, we present a novel boosting approach that leverages cheaper, lower-fidelity data of the problem at hand to identify the *best* sketch among a set of candidate sketches. This in turn specifies the sketch of the intended high-fidelity model and the associated data. We provide theoretical analyses of this bifidelity boosting (BFB) approach and discuss the conditions the low- and high-fidelity data must satisfy for a successful boosting. In doing so, we derive a bound on the residual norm of the BFB sketched solution relating it to its ideal, but computationally expensive, high-fidelity boosted counterpart. Empirical results on both manufactured and PDE data corroborate the theoretical analyses and illustrate the efficacy of the BFB solution in reducing the regression error, as compared to the nonboosted solution.

Key words. randomized sketching, boosting, uncertainty quantification, multifidelity, least squares, polynomial chaos

MSC codes. 68Q25, 68R10, 68U05

DOI. 10.1137/22M1524989

1. Introduction.

In forward uncertainty quantification (UQ) involving computationally expensive models, one often seeks to build emulators or surrogates of the model; popular examples include polynomial chaos (PC) emulators and Gaussian processes. This paper con-

*Received by the editors September 27, 2022; accepted for publication (in revised form) November 13, 2023; published electronically April 4, 2024.
https://doi.org/10.1137/22M1524989
Funding: This work was supported by AFOSR awards FA9550-20-1-0138 and FA9550-20-1-0188 with Dr. Fariba Fahroo as the program manager, and by DOE ASCR MMICC grant DE-SC0023346. The views expressed in the article do not necessarily represent the views of the AFOSR, DOE, or the U.S. Government.

†Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309-0526 USA (nuojin.cheng@colorado.edu, stephen.becker@colorado.edu).
‡Applied Mathematics & Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720 USA (oamalik@lbl.gov).
§Department of Statistics and Actuarial Science, University of Waterloo, Waterloo, ON N2L 3G1, Canada (yiming.xu@uwaterloo.ca).
¶Sead Aerospace Engineering Sciences Department, University of Colorado at Boulder, Boulder, CO 80309 USA (doostan@colorado.edu).
‖Scientific Computing and Imaging Institute, and Department of Mathematics, University of Utah, Salt Lake City, UT 84112 USA (akil@sci.utah.edu).

siders the specialized case when one seeks to build an emulator that predicts the output of a model as a function of its input; in practice the input is frequently a finite-dimensional parameter $\boldsymbol{p}$, which is often modeled as a random vector to account for either uncertainty in precise values of these parameters or as a means to model variability of parameters in order to assess robustness of an output [29, 38]. Once an emulator is built, various statistics of the output (which is random due to the randomness in the input $\boldsymbol{p}$) can be computed by directly manipulating the emulator.

In the simplest setting, observations of the model (that is, possibly noisy input-output pairs) are required to train emulators. To mitigate data collection cost, one wishes to use as few observations or *samples* of the model as possible. In this context, the goal is to engineer a "good" set of samples, which is the purview of complexity theory and optimal experimental design. The main goal of this paper is to develop a new strategy for randomized construction of a sampling design by leveraging access to data from a "low fidelity" model, that is, a second computational model with the same input that is less expensive to query and whose output is a possibly inaccurate approximation to the original model's output. Such "bifidelity" setups are ubiquitous in UQ and are specializations of more general multifidelity scenarios, in which even inaccurate low-fidelity models can contain useful information for high-fidelity prediction [35]. In summary, our proposed procedure combines ideas from sketching of high-fidelity least squares problems with statistical boosting methods employing low-fidelity data. We show that this results in a procedure that can leverage low-fidelity (i.e., inexpensive) data to increase the accuracy of an emulator.

1.1. Problem setup. To formalize concepts, we consider a model $\mathcal{T}$ given by a (possibly nonlinear) parameter-to-output map,

$$b = \mathcal{T}(\boldsymbol{p}), \qquad \boldsymbol{p} \in \Omega \subset \mathbb{R}^q, \qquad \mathcal{T} : \Omega \to \mathbb{R}. \tag{1.1}$$

A canonical example is when $\mathcal{T}$ is a measurement functional (e.g., the spatial average) operating on the solution to an elliptic partial differential equation (PDE) whose formulation contains random variables $\boldsymbol{p}$ that parameterize, e.g., the diffusion coefficient. Hence, $\mathcal{T}$ is the composition of a measurement functional with the solution map of a parametric PDE. By placing a probability distribution on $\boldsymbol{p}$ that reflects a model of uncertainty, the goal of forward UQ is to quantify the resulting randomness in $b(\boldsymbol{p})$, frequently via statistics. Since explicit formulas revealing the dependence of b on $\boldsymbol{p}$ are typically not available, one resorts to approximations via emulators.

In this paper we consider building emulators for forward UQ via a nonintrusive least squares–based strategy. More precisely, we assume an a priori form for an emulator b_V,

$$b(\boldsymbol{p}) \approx b_V(\boldsymbol{p}) := \sum_{j=1}^{d} x_j^* \psi_j(\boldsymbol{p}), \qquad V := \operatorname{span}\{\psi_1, \ldots, \psi_d\}, \tag{1.2}$$

where ψ_j are fixed, known functions (in PC approaches they are multivariate polynomial functions of $\boldsymbol{p}$), and the coefficients x_j^* must be determined. In the context of generalized linear models, we have assumed that covariates ψ_j are identified. We compute the coefficients x_j^* through regression, i.e., we identify x_j^* through data collected from evaluating the expensive computer model b on a prescribed design or ensemble of samples $\{\boldsymbol{p}_n\}_{n=1}^N$. The coefficients x_j^* are then chosen as the solution to a least squares problem,

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}\in\mathbb{R}^d} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2, \qquad \boldsymbol{A}(n,j) = \sqrt{w}_n \psi_j(\boldsymbol{p}_n), \qquad \boldsymbol{b}(n) = \sqrt{w}_n b(\boldsymbol{p}_n), \tag{1.3}$$

where $\boldsymbol{A} \in \mathbb{R}^{N\times d}$ is referred to as the *design matrix* of the problem, and we have introduced positive weights $\{w_n\}_{n=1}^N$ that result in a general weighted least squares problem. In (1.3) we assume (1) that the parameter ensemble is sufficiently dense or space-filling so that $\boldsymbol{x}^*$ is considered as a sufficiently accurate emulator, and (2) that the vector $\boldsymbol{b}$ comprises exact (and not noisy) evaluations of the computational model. In principle one could augment our analysis to consider $\boldsymbol{b}$ as containing noisy evaluations; this does not change theoretical guarantees or practical algorithmic details. However, in this case $\boldsymbol{x}^*$ corresponds to regression coefficients conditioned on the noise realization, with sufficiently small noise so that $\boldsymbol{x}^*$ is still assumed sufficiently accurate for prediction purposes. We reiterate that while $\boldsymbol{b}$ may contain noise, we assume in what follows that it is deterministic for simplicity.

For many sampling designs, such as low-discrepancy sequences, the weights in (1.3) can be taken as uniform; for other sampling designs, such as quadrature rules, the weights w_n are given by the (positive) quadrature weights. Once $\boldsymbol{x}^*$ is computed, the emulator b_V can be manipulated and computationally analyzed to compute (approximate) statistics for b or the sensitivity of b to each entry of $\boldsymbol{p}$. The challenge with this approach is that when $\dim \boldsymbol{p} = q \gg 1$, then designing an ensemble (or quadrature rule) that yields sufficient accuracy typically requires $N \gg 1$ samples of b, which is prohibitively expensive when such evaluations amount to PDE solutions. (For example, if $\boldsymbol{p}$ has independent components then a q-dimensional tensorization of an n-point ensemble in each dimension requires $N = n^q$ points.)

Our proposed strategy mitigates this cost in a bifidelity UQ setup via a procedure that combines statistical boosting with linear sketching (see, e.g., [30, section 7.2] and [40, section 2.3]). *Sketching* refers to a compression of high-dimensional data into a low-dimensional space; we restrict our attention to the popular *linear* sketching techniques, in which case a linear sketch operating on a high-dimensional vector $\boldsymbol{b}$ (such as the vector $\boldsymbol{b}$ in (1.3)) can be represented by the action of a matrix $\boldsymbol{S} \in \mathbb{R}^{m\times N}$ with $m < N$. We then refer to $\boldsymbol{b} \mapsto \boldsymbol{S}\boldsymbol{b}$ as a sketching operation, and $\boldsymbol{S}$ as the corresponding sketch matrix. In our context, the sketch matrix is a random matrix whose characteristics we precisely specify later; see Definition 2.1 and subsections 2.2.2 to 2.2.4. In particular, there are *row sketches* (having nonzero entries in only m columns) for which computing $\boldsymbol{S}\boldsymbol{b}$ requires knowledge of only m entries of $\boldsymbol{b}$, rather than all N entries required for a general dense $\boldsymbol{S}$. (This corresponds to subsampling the full N data points, generally with replacement, but sketching operators that are not row sketches are more general than subsampling.) For a given sketch matrix $\boldsymbol{S}$, there is a corresponding sketched least squares problem that is derived from (1.3) by first applying $\boldsymbol{S}$ to both sides:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in\mathbb{R}^d} \|\boldsymbol{S}\boldsymbol{A}\boldsymbol{x} - \boldsymbol{S}\boldsymbol{b}\|_2^2.$$

One hopes that $\hat{\boldsymbol{x}}$ is "close" to $\boldsymbol{x}^*$. One may wonder why making a deterministic choice of $\boldsymbol{S}$ is not utilized: Without a priori knowledge of $\boldsymbol{b}$, a deterministic sketch with $m < N$ generally is not robust to adversarial vectors $\boldsymbol{b}$ that result in a large residual for $\hat{\boldsymbol{x}}$ relative to the residual for $\boldsymbol{x}^*$. However, in general scenarios one can identify constructive *probabilistic* models for $\boldsymbol{S}$ where sketches of near-optimal size, $m \gtrsim d\log d/(\epsilon\delta)$, ensure

$$\|\boldsymbol{A}\hat{\boldsymbol{x}} - \boldsymbol{b}\|^2 \le (1+\epsilon)\|\boldsymbol{A}\boldsymbol{x}^* - \boldsymbol{b}\|^2 \quad \text{with probability } \ge 1-\delta.$$

Since $\boldsymbol{S}$ is random, then $\hat{\boldsymbol{x}}$ is random, and hence the accuracy of $\hat{\boldsymbol{x}}$ relative to $\boldsymbol{x}^*$ has a random distribution; in particular, the guarantees above reveal the possibility of "failure" events of nonzero probability δ. The idea of *boosting* is to generate and use several, say L, realizations of $\hat{\boldsymbol{x}}$ to identify a "boosted" approximation for $\hat{\boldsymbol{x}}$ with a more favorable accuracy distribution. The most transparent and simple example is to boost by choosing the realization with the highest accuracy among the L realizations. Thus, a naive boosting strategy would generate L i.i.d. samples of $\{\hat{\boldsymbol{x}}_i\}_{i\in[L]}$, and the boosted choice would correspond to

$$\underset{\boldsymbol{y}\in\{\hat{\boldsymbol{x}}_i\}_{i\in[L]}}{\arg\min} \ \|\boldsymbol{A}\boldsymbol{y}-\boldsymbol{b}\|.$$

However, even using row sketches, computing these residuals essentially requires full knowledge of $\boldsymbol{b}$, which we wish to avoid when each component of this vector is an expensive PDE solve. Our approach attacks this problem in the sketch selection boosting phase by replacing $\boldsymbol{b}$ with an approximate, low-fidelity version (that we subsequently introduce as $\widetilde{\boldsymbol{b}}$) from which collecting a large number of samples is computationally feasible. Once a "good" sketch is identified in the boosting phase using low-fidelity (inexpensive) data, we solve a single sketched least squares problem involving high-fidelity (expensive) data $\boldsymbol{b}$. In summary, our approach boosts the randomness of the sketching operator $\boldsymbol{S}$, using a relatively large number (N) of low-fidelity model evaluations in a boosting phase to identify a favorable sketch operator, and subsequently using a relatively small number (m) of high-fidelity model evaluations to compute regression coefficients using the favorable sketch operator.

1.2. Contributions of this article. The contributions of this article are as follows:

- We propose a new bifidelity boosting (BFB) algorithm to compute an approximation to $\boldsymbol{x}^*$. The procedure, given in Algorithm 3.1, computes the solution of a *sketched* least squares problem, where the sketch matrix is identified by a boosting procedure on a low-fidelity data vector $\widetilde{\boldsymbol{b}}$. The sketching approach reduces the required sample complexity from N evaluations of $\boldsymbol{b}$ to $\sim d\log d$ samples of $\boldsymbol{b}$, which can be a significant saving. The boosting procedure requires $\sim Ld\log d$ evaluations of the low-fidelity model $\widetilde{\boldsymbol{b}}$, where, in the language of statistical learning, L is the number of weak learners used in the boosting procedure. When $\widetilde{\boldsymbol{b}}$ costs substantially less than $\boldsymbol{b}$, this cost for collecting the boosting data is negligible.
- We provide a theoretical analysis of BFB under certain assumptions, which provides quantitative bounds on the residual of the BFB solution $\hat{\boldsymbol{x}}_{\mathsf{BFB}}$ relative to the full, computationally expensive solution $\boldsymbol{x}^*$ (see Theorems 3.2 and 3.4). This in particular reveals a relationship between L and a type of correlation between $\boldsymbol{b}$ and $\widetilde{\boldsymbol{b}}$ that provides guidance for when BFB is useful. (See the discussion following Theorem 3.4.) We also provide some asymptotic bounds on the correlation between the low- and high-fidelity solutions in a certain sense (see Theorem 3.5). Finally, we provide concrete computational strategies to ensure that the required assumptions of BFB hold (see Theorem 3.11).
- We investigate the numerical performance of BFB when combined with several different sampling strategies and compare the performance to the corresponding sampling strategies without boosting. We also demonstrate using real-world problems that the assumptions required for BFB's theoretical analysis frequently hold in practice.

The idea of sketching for least squares solutions has a substantial history in the computer science and numerical linear algebra communities [30, 40]. Our use of sparse row sketches of size $\sim d$ is identical to existing methods for leverage score–based [30], Gaussian sketch–based [32], and volume-maximizing sketching [8, 9]. In addition, boosting for least squares problems is also not a new idea [21]. However, our combination of these approaches in a bifidelity setting is new, to the best of our knowledge, and our analysis in this bifidelity context provides novel, nontrivial insight into the algorithm performance.

The rest of this article is organized as follows. Section 2 introduces the notation we use and provides some background material on various sketching approaches in least squares approximation. Section 3 presents the BFB algorithm along with its theoretical analysis. Section 4 contains numerical experiments which illustrate various aspects of the BFB approach. Sections SM1 and SM2 in the supplementary material describe sketching strategies and contain a discussion of the algorithmic sketching approach in [31] used here. Sections SM3 to SM6 contain some proofs of auxiliary technical results.

2. Preliminaries.

This section introduces notation and describes four sketching strategies for the least squares problem (1.3), namely, column-pivoted QR, leverage scores, volume maximization, and Gaussian sketching.

2.1. Notation.

Matrices are denoted by bold uppercase letters (e.g., $\boldsymbol{A}$), vectors are denoted by bold lowercase letters (e.g., $\boldsymbol{x}$), and scalars are denoted by lowercase regular and Greek letters (e.g., a and α). Entries of matrices and vectors are indicated in parentheses. For example, $\boldsymbol{A}(i,j)$ is the entry on position (i,j) in $\boldsymbol{A}$ and $\boldsymbol{a}(i)$ is the ith entry in $\boldsymbol{a}$. A colon is used to denote all entries along a mode of a matrix. For example, $\boldsymbol{A}(i,:)$ is the ith row of $\boldsymbol{A}$ represented as a row vector. For a set of indices $\mathcal{J}$, $\boldsymbol{A}(\mathcal{J},:)$ denotes the submatrix $(\boldsymbol{A}(j,:))_{j\in\mathcal{J}}$ and $\boldsymbol{a}(\mathcal{J})$ denotes the subvector $(\boldsymbol{a}(j))_{j\in\mathcal{J}}$.

The *compact SVD* of a matrix $\boldsymbol{A}$ takes the form $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, where $\boldsymbol{U}$ and $\boldsymbol{V}$ have $\operatorname{rank}(\boldsymbol{A})$ columns and $\boldsymbol{\Sigma}$ is of size $\operatorname{rank}(\boldsymbol{A}) \times \operatorname{rank}(\boldsymbol{A})$. The *pseudoinverse* of $\boldsymbol{A}$ is denoted by $\boldsymbol{A}^\dagger \stackrel{\text{def}}{=} \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T$. For a matrix $\boldsymbol{U}$ with orthonormal columns, we use $\boldsymbol{U}_\perp$ to denote an orthonormal complement of $\boldsymbol{U}$, i.e., $\boldsymbol{U}_\perp$ is any matrix such that $[\boldsymbol{U},\ \boldsymbol{U}_\perp]$ is square and has orthonormal columns. We use $\boldsymbol{P}_{\boldsymbol{A}} \stackrel{\text{def}}{=} \boldsymbol{A}\boldsymbol{A}^\dagger = \boldsymbol{U}\boldsymbol{U}^T$ to denote the *orthogonal projection* onto $\operatorname{range}(\boldsymbol{A})$, where $\boldsymbol{U} = \operatorname{orth}(\boldsymbol{A})$ is a(ny) matrix whose columns are an orthonormal basis for $\operatorname{range}(\boldsymbol{A})$, e.g., via the compact SVD or QR decomposition of $\boldsymbol{A}$. The determinant of $\boldsymbol{A}$ is denoted by $\det(\boldsymbol{A})$. For a positive integer n, we use the notation $[n] \stackrel{\text{def}}{=} \{1, 2, \ldots, n\}$. We use $\boldsymbol{a}_\mathcal{P}$ to denote a vector $\boldsymbol{a} \neq \boldsymbol{0}$ rescaled to unit length, $\boldsymbol{a}_\mathcal{P} = \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_2}$.

We also introduce two notions of correlation: for given deterministic vectors $\boldsymbol{a}, \boldsymbol{b} \neq \boldsymbol{0}$, we define the correlation between them as the cosine of the angle separating them: $\operatorname{corr}(\boldsymbol{a},\boldsymbol{b}) \stackrel{\text{def}}{=} \frac{\langle \boldsymbol{a},\boldsymbol{b}\rangle}{\|\boldsymbol{a}\|_2\|\boldsymbol{b}\|_2}$, where $\langle\cdot,\cdot\rangle$ denotes the Euclidean inner product. We will also require Pearson's correlation coefficient, which is widely used in statistics. For two (nonconstant) random variables X and Y with bounded second moments defined on the same probability space, their correlation is defined as $\operatorname{corr}(X,Y) \stackrel{\text{def}}{=} \frac{\mathbb{E}[(X-\mathbb{E}[X])(Y-\mathbb{E}[Y])]}{\sqrt{\mathbb{V}[X]\mathbb{V}[Y]}}$, where $\mathbb{E}[\cdot]$ and $\mathbb{V}[\cdot]$ are, respectively, the mathematical expectation and variance operators. Note that our notation $\operatorname{corr}(\cdot,\cdot)$ is overloaded, operating differently on vectors and (random) scalars. The context of use in what follows should make it clear which definition above is used.

The following denotes the minimum of the least squares objective in (1.3):

$$r(\boldsymbol{A},\boldsymbol{b}) \stackrel{\text{def}}{=} \min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}\|_2 = \|\boldsymbol{A}\boldsymbol{x}^*-\boldsymbol{b}\|_2, \tag{2.1}$$

where $\boldsymbol{x}^*$ is defined as in (1.3).

Finally, we assume the availability of and leverage a *low-fidelity* model $\widetilde{b}(\boldsymbol{p})$ that is an approximation to $b(\boldsymbol{p})$ defined in (1.1). For example, $\widetilde{\boldsymbol{b}}$ may correspond to using a discretized PDE solver with a mesh coarser than the one which produces accurate realizations of $\boldsymbol{b}$, or to model approximations such as Reynolds-averaged Navier–Stokes solvers, or to solutions computed with arithmetic in lower precision compared to samples for $\boldsymbol{b}$. Although $\widetilde{\boldsymbol{b}}$ may be untrusted as a replacement for $\boldsymbol{b}$, it can be used to extract some useful information about $\boldsymbol{b}$, as is done in the now standard multifidelity approaches [35]. Throughout this paper, we assume the bifidelity setup, i.e., two levels of fidelity, and also that the cost of evaluating $\widetilde{\boldsymbol{b}}$ is much less than the corresponding cost for $\boldsymbol{b}$; both of these are common practical assumptions [11, 33, 42, 18, 34].

2.2. Sketching of least squares problems. Solving problem (1.3) using standard methods (e.g., via the QR decomposition) costs $\mathcal{O}(Nd^2)$.[1] When N is large, this may be prohibitively expensive. A popular approach to address this issue is to apply a *sketch operator* $\boldsymbol{S}\in\mathbb{R}^{m\times N}$, where $m \ll N$ to both $\boldsymbol{A}$ and $\boldsymbol{b}$ in (1.3) in order to reduce the size of the problem:

$$\hat{\boldsymbol{x}} \stackrel{\text{def}}{=} \operatorname*{arg\,min}_{\boldsymbol{x}\in\mathbb{R}^d} \|\boldsymbol{S}\boldsymbol{A}\boldsymbol{x}-\boldsymbol{S}\boldsymbol{b}\|_2 . \tag{2.2}$$

This approach has two benefits: (i) If $\boldsymbol{S}$ is a row sketch, i.e., has only a small number of nonzero columns, then $\boldsymbol{S}\boldsymbol{b}$ requires knowledge of only a small number of entries of $\boldsymbol{b}$, and (ii) the cost of solving this smaller problem is $\mathcal{O}(md^2)$, a substantial reduction from $\mathcal{O}(Nd^2)$ when $m \ll N$. Analogously to (2.1), we will use the following to denote the least squares objective value for the approximate solution:

$$r_{\boldsymbol{S}}(\boldsymbol{A},\boldsymbol{b}) \stackrel{\text{def}}{=} \|\boldsymbol{A}\hat{\boldsymbol{x}}-\boldsymbol{b}\|_2. \tag{2.3}$$

The goal is for the approximation $\hat{\boldsymbol{x}}$ to yield a residual "close" to the optimal residual of the full problem (1.3), i.e., $r(\boldsymbol{A},\boldsymbol{b}) \approx r_{\boldsymbol{S}}(\boldsymbol{A},\boldsymbol{b})$, which is achieved if m is "large enough." The following definition makes this more precise.

Definition 2.1 ((ε,δ) pair condition). *Let $\boldsymbol{S}\in\mathbb{R}^{m\times N}$ be a random matrix. Given $\boldsymbol{A}\in\mathbb{R}^{N\times d}$, $\boldsymbol{b}\in\mathbb{R}^N$, and $\varepsilon,\delta>0$, the distribution of $\boldsymbol{S}$ is said to satisfy an (ε,δ) pair condition for $(\boldsymbol{A},\boldsymbol{b})$ if, with probability at least $1-\delta$, both conditions*

$$\operatorname{rank}(\boldsymbol{S}\boldsymbol{A}) = \operatorname{rank}(\boldsymbol{A}) \qquad \text{and} \qquad r_{\boldsymbol{S}}(\boldsymbol{A},\boldsymbol{b}) \le (1+\varepsilon)\, r(\boldsymbol{A},\boldsymbol{b}) \tag{2.4}$$

hold simultaneously, where $r(\boldsymbol{A},\boldsymbol{b})$ and $r_{\boldsymbol{S}}(\boldsymbol{A},\boldsymbol{b})$ are defined as in (2.1) and (2.3), respectively.

Such a condition can be satisfied with $m < N$ samples; sections 2.2.2, 2.2.3, and 2.2.4 provide three explicit examples satisfying (2.4) that we employ in our simulations. We also

[1] In our context, we have $N \ge d$; see Assumption 3.1.

present a deterministic sketching strategy in subsection 2.2.1 that is of practical interest due to its simplicity and effectiveness, but does not satisfy (2.4).

In particular, sketching operators $\boldsymbol{S}$ that sample a subset of the rows are of particular interest in UQ since $\boldsymbol{Sb}$ in (2.2) then requires knowledge of only a subset of entries in the vector $\boldsymbol{b}$, meaning that fewer samples need to be collected. Our sketches in subsections 2.2.1 to 2.2.3 are row subsampling sketches, whereas the Gaussian sketch in subsection 2.2.4 samples all rows of $\boldsymbol{b}$. We are very brief in our discussion since these sketching techniques are well known (see, e.g., [23, 30, 40, 32]), and refer the reader to section SM1 for a more detailed discussion of known guarantees for random sketches.

2.2.1. Sampling via column-pivoted QR decomposition. The following is a deterministic and heuristic method for defining a sketching operator that corresponds to subselecting rows (without replacement) of the least squares problem. Let $\boldsymbol{A}^T\boldsymbol{P} = \boldsymbol{A}(\mathcal{J},:)^T = \boldsymbol{QR}$ be a column-pivoted QR (CPQR) decomposition where $\mathcal{J}$ is a length-N permutation vector. Choosing the first m rows as indices from the permutation vector $\mathcal{J}$ has been explored to subsample from either tensor product quadratures [37] or random samples (approximate D-optimal design) [22, 10, 20].

However, $\boldsymbol{A}$ is an $N \times d$ matrix and so when $m > d$, the entries $\mathcal{J}(d+1:N)$ have no particular meaning (i.e., can be arbitrarily ordered). Our heuristic to circumvent this issue is to remove rows $\mathcal{J}(1:m)$ from $\boldsymbol{A}$, and to subsequently perform another CPQR on the resulting $(N-m)\times d$ submatrix of $\boldsymbol{A}$, which can provide another m meaningful indices. One can repeat this process until d total indices have been chosen. The formal procedure along with some extra discussion is provided in Algorithm SM1.1 and subsection SM1.1, respectively.

This approach, being deterministic, cannot satisfy guarantees in Definition 2.1, but if $m = d$, one can prove bounds on the condition number of $\boldsymbol{A}(\mathcal{J}(1:d),:)$; see [37, Lemma 2.1].

2.2.2. Leverage score sampling. Let $\boldsymbol{A} = \boldsymbol{U\Sigma V}^T$ be a compact SVD; the *leverage scores* ℓ_i of $\boldsymbol{A}$, and corresponding normalized probability values p_i are defined as

$$\ell_i(\boldsymbol{A}) \overset{\text{def}}{=} \|\boldsymbol{U}(i,:)\|_2^2, \qquad p_i(\boldsymbol{A}) \overset{\text{def}}{=} \frac{\ell_i(\boldsymbol{A})}{\sum_{i\in[N]}\ell_i(\boldsymbol{A})} \quad \text{for } i \in [N]. \tag{2.5}$$

The matrix $\boldsymbol{U}$ can be replaced with any matrix whose columns form an orthonormal basis for range($\boldsymbol{A}$) [40, section 2.4]. Let $\{F_j\}_{j\in[m]}$ be an i.i.d. collection of $[N]$-valued random variables such that $\mathbb{P}\{F_j = i\} = p_i(\boldsymbol{A})$. The leverage score sampling sketch $\boldsymbol{S} \in \mathbb{R}^{m\times N}$ is defined elementwise via

$$\boldsymbol{S}_{ji} = \frac{\text{Ind}\{F_j = i\}}{\sqrt{m p_{F_j}(\boldsymbol{A})}} \quad \text{for } (j,i) \in [m]\times[N], \tag{2.6}$$

where Ind$\{A\}$ is the indicator function which is 1 if the random event A occurs and zero otherwise. Algorithms and theory for leverage score sampling have been developed in a number of papers; see, e.g., [13, 14, 15, 30, 28] and references therein. The distribution for the leverage score sketch in (2.6) satisfies an (ε,δ) condition for $(\boldsymbol{A},\boldsymbol{b})$ if $m \gtrsim d\log(d/\delta) + d/(\varepsilon\delta)$; see Theorem 3.11 for a more detailed and slightly stronger statement, and subsection SM1.2 for more discussion on leverage scores.

Computing a matrix $\boldsymbol{U}$ used to define leverage score sampling can be expensive when $\boldsymbol{A}$ is large—our simulations utilize an algorithm for quickly and exactly computing leverages

for certain types of structured matrices [31]. Leverage score sampling is a row subsampling procedure, with replacement.

2.2.3. Leveraged volume sampling. Fixing m, volume sampling chooses a row subset of size m according to a distribution where the probability of choosing index set $\mathcal{J} \subset [N]$ is $\mathbb{P}(\mathcal{J}) \propto \det(\boldsymbol{A}(\mathcal{J},:)^T \boldsymbol{A}(\mathcal{J},:))$; see, e.g., [7, 8]. That is, volume sampling is a determinantal point process. Volume sampling algorithms have at-best linear-in-N complexity, which can be prohibitive in quadrature sampling since N can be very large.

Leveraged volume sampling [9] augments the standard volume sampling procedure and results in N-independent sampling algorithms with a sketch distribution that satisfies an (ε, δ) condition for $(\boldsymbol{A}, \boldsymbol{y})$ if $m \gtrsim d\log(d/\delta) + d/(\varepsilon\delta)$, like leverage score sampling. We implement leveraged volume sampling using a combination of the techniques in [8, 31]. See subsection SM1.3 for a more precise algorithmic description along with a complexity discussion. Like leverage score sampling, this procedure subsamples rows with replacement.

2.2.4. Gaussian sketching operator. The Gaussian sketching operator $\boldsymbol{S} \in \mathbb{R}^{m\times N}$ has entries that are i.i.d. Gaussian random variables with mean zero and variance $1/m$. The Gaussian sketch satisfies an (ε, δ) condition if $m \gtrsim (d/\varepsilon)\log(d/\delta)$. These results also extend to the case when the entries of $\boldsymbol{S}$ are sub-Gaussian; see Theorem 3.11 for further details.

The main benefit of the Gaussian sketching operator is that it allows for simple and precise theoretical analysis of procedures that use sketching as a subroutine [32, Remark 8.2]. This is our motivation for considering the Gaussian sketch in this paper. Computationally, it is not efficient to use Gaussian sketching for least squares problems: Computing $\boldsymbol{SA}$ costs $\mathcal{O}(mNd)$, which is more than the $\mathcal{O}(Nd^2)$ cost of solving the original least squares problem (recall that $m > d$). Additionally, in bifidelity estimation computing $\boldsymbol{Sb}$ requires knowledge of all elements of $\boldsymbol{b}$, which is prohibitively expensive when that vector contains high-fidelity data.

2.3. Bifidelity problems. The main goal of this paper is to propose a strategy that improves the accuracy of sketching via a boosting procedure that employs a full vector $\tilde{\boldsymbol{b}}$ corresponding to an inexpensive low-fidelity approximation to $\boldsymbol{b}$.

Bifidelity frameworks assume the availability of a low-fidelity simulation $\widetilde{\mathcal{T}}$; that is, a map $\widetilde{\mathcal{T}} : \mathbb{R}^q \to \mathbb{R}$ such that $\widetilde{\mathcal{T}}$ is parametrically "correlated" with $\mathcal{T}$ in some sense, but need not be close to $\mathcal{T}$ in terms of sampled values. Such properties arise, for example, in parametric PDE contexts when $\widetilde{\mathcal{T}}$ results from a discretized PDE solution operator on a spatial mesh that is coarser (and hence less trusted) than the mesh corresponding to $\mathcal{T}$. The decreased accuracy/trustworthiness of $\widetilde{\mathcal{T}}$ is balanced by its decreased cost, so that employment of $\widetilde{\mathcal{T}}$ may not furnish precise high-fidelity information, but may provide useful knowledge in terms of dependence on the parameter $\boldsymbol{p}$ with substantially reduced cost.

In the context of constructing our emulator (1.3), our core assumption is that the low-fidelity operator $\widetilde{\mathcal{T}}$ is cheap enough so that exploration of the response over the full ensemble $\{\boldsymbol{p}_i\}_{i\in[N]}$ is more computationally feasible, resulting in a vector $\tilde{\boldsymbol{b}} \in \mathbb{R}^N$ defined as

$$\tilde{\boldsymbol{b}}(n) = \sqrt{w_n}\widetilde{\mathcal{T}}(\boldsymbol{p}_n). \tag{2.7}$$

Of course, one may propose constructing the emulator $\mathcal{T}$ in (1.3) by simply replacing $\boldsymbol{b}$ by $\tilde{\boldsymbol{b}}$, but this restricts the accuracy of the emulator $\mathcal{T}$ to the potentially bad accuracy of $\widetilde{\mathcal{T}}$. In this

paper, we propose a more sophisticated use of $\tilde{\boldsymbol{b}}$, in conjunction with a single sparse sketch of $\boldsymbol{b}$, that retains some accuracy characteristics of $\boldsymbol{x}^*$.

3. Bifidelity boosting in sketched least squares problems. In practice, one often requires the probability of successfully obtaining a good approximation $\boldsymbol{x}^*$ associated with a random sketch from section 2.2 to be sufficiently close to 1, and one way to achieve this with fixed sketch size is through a boosting procedure. With L sketching matrices $\{\boldsymbol{S}_\ell \in \mathbb{R}^{m\times N}\}_{\ell\in[L]}$, one computes the residual for the $\boldsymbol{S}_\ell$-sketched solution (i.e., $\|\boldsymbol{A}(\boldsymbol{S}_\ell\boldsymbol{A})^\dagger(\boldsymbol{S}_\ell\boldsymbol{b}) - \boldsymbol{b}\|_2$) for each $\boldsymbol{S}_\ell$ and then selects the one that yields the smallest residual for use. Even if each sketch sparsely samples rows, this straightforward procedure inflates the required sampling cost of the forward model $\mathcal{T}$ by the factor L, which may be computationally prohibitive. To ameliorate this boosting cost, we employ a bifidelity strategy.

In section 3.1 we present our proposed bifidelity boosting algorithm. Sections 3.2 and 3.3 give our preasymptotic and asymptotic analysis results, respectively. We collect some preliminary technical results in section 3.4 and prove our preasymptotic results in section 3.5. The asymptotic result is proven in section SM5. Section 3.6 provides results for random sketches achieving the (ϵ, δ) condition in Definition 2.1.

3.1. Proposed algorithm. We propose a modified boosting procedure, *bifidelity boosting* (BFB), where the boosting phase of a sketched least squares problem replaces high-fidelity data with low-fidelity data to find the "best" sketching operator and then employs this sketch to compute an approximate least squares solution with high-fidelity data.

Recall that full information of the high-fidelity data $\boldsymbol{b}$ is unaffordable to collect, but full information about a low-fidelity approximation $\tilde{\boldsymbol{b}}$ may be feasible to collect, and the bifidelity procedure is sensible if $\boldsymbol{b}$ is "correlated" in some sense with $\tilde{\boldsymbol{b}}$ (this is codified via the parameter ν introduced later in Theorem 3.2). The BFB procedure is outlined in Algorithm 3.1.

Algorithm 3.1. Bifidelity boosting (BFB).

Input: design matrix $\boldsymbol{A}$, low-fidelity vector $\tilde{\boldsymbol{b}}$, method for computing entries of the high-fidelity vector $\boldsymbol{b}$, collection of sketches for boosting $\{\boldsymbol{S}_\ell\}_{\ell\in[L]}$

Output: an approximate solution $\hat{\boldsymbol{x}}_{\mathsf{BFB}}$ to (1.3)

1: **for** $\ell \in [L]$ **do**

2: compute the ℓth sketched solution $\hat{\boldsymbol{x}}_\ell$ using the low-fidelity data:

$$\hat{\boldsymbol{x}}_\ell = \underset{\boldsymbol{x}\in\mathbb{R}^d}{\arg\min} \left\| \boldsymbol{S}_\ell\boldsymbol{A}\boldsymbol{x} - \boldsymbol{S}_\ell\tilde{\boldsymbol{b}} \right\|_2 \tag{3.1}$$

3: **end for**

4: find the best low-fidelity sketch index ℓ^* using boosting:

$$\ell^* = \underset{\ell\in[L]}{\arg\min} \|\boldsymbol{A}\hat{\boldsymbol{x}}_\ell - \tilde{\boldsymbol{b}}\|_2 \tag{3.2}$$

5: use sketch $\boldsymbol{S}_{\ell^*}$ to compute an approximate solution to (1.3):

$$\hat{\boldsymbol{x}}_{\mathsf{BFB}} = \underset{\boldsymbol{x}\in\mathbb{R}^d}{\arg\min} \|\boldsymbol{S}_{\ell^*}\boldsymbol{A}\boldsymbol{x} - \boldsymbol{S}_{\ell^*}\boldsymbol{b}\|_2 \quad \text{(Requires computing } m \text{ entries of } \boldsymbol{b}) \tag{3.3}$$

The oracle sketch in this scenario is the one identified by the boosting strategy operating directly on the high-fidelity least squares problem, which is computationally unaffordable:

$$\ell^{**} = \arg\min_{\ell\in[L]} \|\boldsymbol{A}\hat{\boldsymbol{x}}_\ell - \boldsymbol{b}\|_2^2, \quad \text{where } \hat{\boldsymbol{x}}_\ell = \arg\min_{\boldsymbol{x}\in\mathbb{R}^d} \|\boldsymbol{S}_\ell\boldsymbol{A}\boldsymbol{x} - \boldsymbol{S}_\ell\boldsymbol{b}\|_2. \tag{3.4}$$

In the coming sections we will theoretically investigate the sketch transferability between high- and low-fidelity boosting, i.e., when the residual associated to $\hat{\boldsymbol{x}}_{\mathsf{BFB}}$, the solution produced by Algorithm 3.1, is comparable to the residual associated to $\hat{\boldsymbol{x}}_{\ell^{**}}$.

We divide our analysis into two cases: Our first analysis frames performance of Algorithm 3.1 in terms of an *optimality coefficient,* defined in (3.5), which measures the quality of the least squares residual for a particular sketch $\boldsymbol{S}$; we provide preasymptotic analysis (see Theorem 3.2) with quantitative results that provides qualitative guidance on how the BFB algorithm behaves in terms of the tradeoff in the number of sketches L versus a correlation parameter (see the discussion following Theorem 3.4). We also provide asymptotic analysis with Gaussian sketches that confirms intuition that the probabilistic correlations between the low- and high-fidelity random sketches is high when $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ have high vector correlations (see the discussion around Theorem 3.5). For analysis purposes we make the following assumption.

Assumption 3.1. Assume that neither $\tilde{\boldsymbol{b}}$ nor $\boldsymbol{b}$ lies in range$(\boldsymbol{A})$, i.e., $\tilde{\boldsymbol{b}}, \boldsymbol{b} \notin \text{range}(\boldsymbol{A})$.

This is a reasonable assumption: If $\boldsymbol{b} \in \text{range}(\boldsymbol{A})$, then it would be possible to solve the high-fidelity least squares problem exactly by sampling $m = d$ linearly independent rows of $\boldsymbol{A}$ and the corresponding rows of $\boldsymbol{b}$. In this case, it is therefore easy to solve (1.3) and only requires accessing d rows of $\boldsymbol{b}$. Similarly, if $\tilde{\boldsymbol{b}} \in \text{range}(\boldsymbol{A})$, then it would be easy to compute a sketch $\boldsymbol{S}_\ell$ which only samples $m = d$ rows and achieves zero error in line 4 of Algorithm 3.1, therefore making the boosting procedure vacuous.

3.2. Preasymptotic analysis via optimality coefficients.

We introduce the following measure of relative error difference between the sketched and optimal solutions:

$$\mu_{\boldsymbol{A}}(\boldsymbol{b}, \boldsymbol{S}) \overset{\text{def}}{=} \sqrt{\frac{r_{\boldsymbol{S}}^2(\boldsymbol{A},\boldsymbol{b}) - r^2(\boldsymbol{A},\boldsymbol{b})}{r^2(\boldsymbol{A},\boldsymbol{b})}} \overset{(*)}{=} \frac{\|(\boldsymbol{S}\boldsymbol{Q})^\dagger \boldsymbol{S}\boldsymbol{Q}_\perp\boldsymbol{Q}_\perp^T\boldsymbol{b}\|_2}{\|\boldsymbol{Q}_\perp\boldsymbol{Q}_\perp^T\boldsymbol{b}\|_2}, \tag{3.5}$$

where $\boldsymbol{Q} = \text{orth}(\boldsymbol{A})$, and the second equality, marked $(*)$, is valid if $\text{rank}(\boldsymbol{S}\boldsymbol{A}) = \text{rank}(\boldsymbol{A})$, which we establish in Lemma 3.7. For notational simplicity we usually write $\mu(\boldsymbol{b}, \boldsymbol{S})$ when $\boldsymbol{A}$ is clear from the context, but we emphasize that μ does depend on $\boldsymbol{A}$. Note that $r(\boldsymbol{A},\boldsymbol{b}) = \|\boldsymbol{Q}_\perp\boldsymbol{Q}_\perp^T\boldsymbol{b}\|_2 > 0$ due to Assumption 3.1, so the denominator in (3.5) is nonzero. We call μ the *optimality coefficient.* Smaller values of μ are better in practice: $\mu = 0$ implies the sketch achieves perfect reconstruction of the data relative to the full least squares solution.

We provide two main theoretical results, Theorems 3.2 and 3.4, which shed light on the performance of Algorithm 3.1 from two different practical perspectives. Theorem 3.2 shows that with an appropriate choice of the sketches $\{\boldsymbol{S}_\ell\}_{\ell\in[L]}$, Algorithm 3.1 produces a solution with a relative error close to that of the oracle sketch solution in (3.4), while Theorem 3.4 provides insight on constructing an appropriate sketch for the purpose of bifidelity boosting.

A discussion of the practical implications of these two theorems is presented at the end of section 3.2. Note that it would be straightforward to provide such guarantees if $r_{\boldsymbol{S}}(\boldsymbol{A},\tilde{\boldsymbol{b}}) \le r_{\boldsymbol{S}'}(\boldsymbol{A},\tilde{\boldsymbol{b}})$ implied $r_{\boldsymbol{S}}(\boldsymbol{A},\boldsymbol{b}) \le r_{\boldsymbol{S}'}(\boldsymbol{A},\boldsymbol{b})$, in which case $\ell^* = \ell^{**}$. This may happen, for instance, when $\tilde{\boldsymbol{b}}$ and $\boldsymbol{b}$ are proportional. Unfortunately, this monotone residual property replacing $\boldsymbol{b}$ with $\tilde{\boldsymbol{b}}$ is unlikely to hold in practice. Theorem 3.2 identifies alternative conditions that ensure $\boldsymbol{S}_{\ell^*}$ is a "good" sketch for the high-fidelity data.

Theorem 3.2. *Fix a positive integer L and suppose $\delta, \varepsilon \in (0,1]$. If $\{\boldsymbol{S}_\ell\}_{\ell\in[L]}$ is a sequence of i.i.d. random matrices whose distribution is an $(\varepsilon, \frac{\delta}{L})$ pair for $(\boldsymbol{Q},\boldsymbol{h})$, where*

$$\boldsymbol{h} \overset{\text{def}}{=} \left((\boldsymbol{P}_{\boldsymbol{Q}_\perp}\boldsymbol{b})_{\mathcal{P}} - (\boldsymbol{P}_{\boldsymbol{Q}_\perp}\tilde{\boldsymbol{b}})_{\mathcal{P}}\right)_{\mathcal{P}} \quad \text{and} \quad \boldsymbol{Q} \overset{\text{def}}{=} \operatorname{orth}(\boldsymbol{A}), \tag{3.6}$$

then with probability at least $1-\delta$,

$$\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^*}) \le \mu(\boldsymbol{b},\boldsymbol{S}_{\ell^{**}}) + 2\sqrt{6(1-\nu)\varepsilon}, \qquad \nu \overset{\text{def}}{=} \left|\operatorname{corr}(\boldsymbol{P}_{\boldsymbol{Q}_\perp}\boldsymbol{b}, \boldsymbol{P}_{\boldsymbol{Q}_\perp}\tilde{\boldsymbol{b}})\right|, \tag{3.7}$$

where ν is the absolute correlation coefficient between $\boldsymbol{P}_{\boldsymbol{Q}_\perp}\boldsymbol{b}$ and $\boldsymbol{P}_{\boldsymbol{Q}_\perp}\tilde{\boldsymbol{b}}$. In addition, in the event that (3.7) is true, we have that (2.4) holds with $\boldsymbol{S} = \boldsymbol{S}_\ell$ for every $\ell \in [L]$.

We prove Theorem 3.2 in section 3.5. Theorem 3.2 shows that if a sketch satisfies an $(\varepsilon, \delta/L)$ condition for the pair $\boldsymbol{Q}$ and an element $\boldsymbol{h}$ of range$(\boldsymbol{Q}_\perp)$, then we are able to prove bounds on the low-fidelity boosted optimality coefficient $\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^*})$ relative to the oracle high-fidelity boosted optimality coefficient $\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^{**}})$. This is quite a general statement that accommodates a wide range of sketching operators. The condition on the operators $\{\boldsymbol{S}_\ell\}_{\ell\in[L]}$ is, for example, satisfied by all sketching operators in sections 2.2.2–2.2.4 when the embedding dimension m is sufficiently large. More precise statements for the leverage score and Gaussian sketches are provided in Theorem 3.11.

To achieve a good approximate solution when applying sketching techniques in least squares problems the sketching operator must preserve the relevant geometry of the problem. In particular, it is key that $\boldsymbol{Q}$ and $\boldsymbol{P}_{\boldsymbol{Q}_\perp}\boldsymbol{b}$ remain roughly orthogonal after the sketching operator is applied. This importance of preserving $\boldsymbol{P}_{\boldsymbol{Q}_\perp}\boldsymbol{b}$ in the sketching phase when $\boldsymbol{b}$ is replaced by low-fidelity data $\tilde{\boldsymbol{b}}$ manifests in Theorem 3.2 through the correlation parameter ν.

Remark 3.3. Equation (3.7) suggests that $\boldsymbol{S}_{\ell^*}$ is "good" when ν is large. This explicitly requires high parametric correlation between the portions of $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ that lie orthogonal to the range of $\boldsymbol{A}$. A more subtle sufficient condition ensuring large ν is furnished by our discussion following Proposition 3.8, which provides a lower bound for ν in terms of other parameters.

Although Theorem 3.2 presents a form of error bound using appropriate sketches, it does not provide a concrete strategy for how the sketches used in boosting are chosen or constructed. Combining Theorem 3.2 with good sketching choices should result in explicit and illuminating theory for Algorithm 3.1. In particular, one expects a tradeoff between the values of ν and L: larger values of ν (close to 1) suggest that the BFB procedure will be more effective (since $\boldsymbol{b}$ is "closer" to $\tilde{\boldsymbol{b}}$), so we expect that larger L is effective in this case. Smaller values of ν suggest

that boosting with large L will not achieve anything better than a nonboosting strategy. The theory we develop below reveals this tradeoff for leverage score sketches in subsection 2.2.2.

Theorem 3.4. *Let $\delta, \epsilon \in (0, 1/2)$ and $L \in \mathbb{N}$ be chosen, and assume that m, d satisfy*

$$d \le \frac{\delta}{4} \exp\left(\frac{2}{35\epsilon\delta}\right), \qquad m \ge \frac{4dL}{\epsilon\delta}. \tag{3.8}$$

Now consider Algorithm 3.1, where $\{\boldsymbol{S}_\ell\}_{\ell\in[L]}$ are i.i.d. samples of the leverage score sketching operator defined in (2.6). Then each $\boldsymbol{S}_\ell$ satisfies an $(\epsilon/L, \delta/2)$ condition for the pair $(\boldsymbol{Q}, \boldsymbol{h})$, and with probability at least $1-\delta$, we have

$$r^2_{\boldsymbol{S}_{\ell^*}}(\boldsymbol{A}, \boldsymbol{b}) \le \left(1 + \frac{\epsilon}{L}\tau\right) r^2(\boldsymbol{A}, \boldsymbol{b}), \tag{3.9}$$

where

$$\tau = \tau(\epsilon, \delta, \nu, L) = 24L(1-\nu) + \frac{\delta}{2}\left(1 + 4\sqrt{6(1-\nu)\epsilon}\right).$$

The results above give explicit behavior of the BFB residual via a concrete sketching strategy for Algorithm 3.1. Note in particular that the sampling requirement $m = \mathcal{O}(L/\epsilon)$ in (3.8) means that *without* boosting and simply generating one sketch $\boldsymbol{S}$ according to (3.8), which requires m high-fidelity samples (equivalent to the number from BFB), we expect that the residual from this one sketch behaves like

$$r^2_{\boldsymbol{S}}(\boldsymbol{A}, \boldsymbol{b}) \sim \left(1 + \frac{\epsilon}{L}\right) r^2(\boldsymbol{A}, \boldsymbol{b}).$$

Comparing the above to (3.9), note that the only difference is the appearance of τ, and hence we expect BFB to be useful (compared to an equivalent number of high-fidelity samples devoted to a nonboosting strategy) when $\tau \le 1$, which requires

$$L \lesssim \frac{1}{1-\nu},$$

i.e., boosting with L sketches is useful in BFB up to a threshold $\sim 1/(1-\nu)$. Boosting with *more* than this threshold level of sketches causes the error bound to saturate at a level determined by $1-\nu$. Since ν is the correlation between the range($\boldsymbol{A}$)-orthogonal components of $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$, we conclude that highly correlated range-orthogonal residuals (large values of ν very close to 1) are optimal for BFB in the sense that sketching with large L will be effective.

A second observation we make is that the $m \sim L$ sampling requirement in (3.8) is larger than necessary: Stronger coherence-like conditions on the matrix $\boldsymbol{A}$ imply that leverage score sketching with $m \sim \log L$ is sufficient; see (3.26) in Theorem 3.11. Gaussian sketches generally only require $m \sim \log L$ samples (see (3.23)). Finally, one can achieve the (ϵ, δ) condition *on average* with leverage scores using $m \sim \log L$ samples (see, e.g., [31, equation (2.18)]. Finally, if d is sufficiently large to violate (3.8), then indeed $m \sim \log L$ for leverage score sketches (see (3.24) and the computation in (SM4.1)). Thus, we expect in practice that $m \sim \log L$ samples are sufficient.

We give the proof of Theorem 3.4 in section SM4, which relies directly on Theorem 3.2, and can be generalized for non–leverage score types of sketches. The theoretical analysis from Theorems 3.2 and 3.4 suggest several practical implications:

- An effective low-fidelity model should have outputs $\tilde{\boldsymbol{b}}$ whose orthogonal projection onto the subspace spanned by the columns of matrix $\boldsymbol{A}$ closely aligns with the same projection of the high-fidelity output vector $\boldsymbol{b}$. This alignment presents the opportunity for mitigating residual errors by increasing the size of the boosting factor L.
- In practice, the choice of the boosting factor L is influenced by the cost of generating low-fidelity data. Larger pools of such data lead to improved boosting, which in turn contribute to BFB error reduction.
- A favorable performance outcome of the BFB is anticipated when a strong "connection" exists between $\tilde{\boldsymbol{b}}$ and $\boldsymbol{b}$, manifesting as a small value for ν as defined in (3.7). Additionally, this favorable performance is further facilitated by increasing the boosting factor L.

3.3. Asymptotic analysis via probabilistic correlation. We provide an alternative analysis of Algorithm 3.1 motivated by the following intuition: If $\mu(\boldsymbol{b}, \boldsymbol{S})$ and $\mu(\tilde{\boldsymbol{b}}, \boldsymbol{S})$ are probabilistically correlated in some sense, then we expect that Algorithm 3.1 should produce a sketching operator $\boldsymbol{S}_{\ell^*}$ that is close to the oracle sketch $\boldsymbol{S}_{\ell^{**}}$. We give a technical verification of this intuition below in Theorem 3.5, providing an asymptotic lower bound on a certain measure of correlation between the two optimality coefficients when $\boldsymbol{S}$ is a Gaussian sketching operator.

Theorem 3.5. *If $\boldsymbol{S}$ is a Gaussian sketch, then*

$$\liminf_{m\to\infty} \operatorname{corr}(\mu^2(\boldsymbol{b}, \boldsymbol{S}), \mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S})) \geq \frac{\|\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b}_{\mathcal{P}}\|_2^2 - \sqrt{6} \min\{\|\boldsymbol{P}_{\boldsymbol{Q}_\perp}(\boldsymbol{b}_{\mathcal{P}} \pm \tilde{\boldsymbol{b}}_{\mathcal{P}})\|_2\}}{\|\boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}}_{\mathcal{P}}\|_2^2}, \tag{3.10}$$

where $\boldsymbol{b}_{\mathcal{P}}, \tilde{\boldsymbol{b}}_{\mathcal{P}}$ are normalized versions of $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$, respectively, and the minimum is taken over the two $\pm$ options. Moreover, if

$$\varphi \overset{\text{def}}{=} \frac{|\langle \boldsymbol{b}, \tilde{\boldsymbol{b}} \rangle|}{\|\boldsymbol{b}\|_2 \|\tilde{\boldsymbol{b}}\|_2} \geq \frac{\|\boldsymbol{P}_{\boldsymbol{Q}} \boldsymbol{b}\|_2}{\|\boldsymbol{b}\|_2} \overset{\text{def}}{=} \kappa, \tag{3.11}$$

then we further have that

$$\liminf_{m\to\infty} \operatorname{corr}(\mu^2(\boldsymbol{b}, \boldsymbol{S}), \mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S})) \geq (1 - \kappa^2) - \frac{\sqrt{12(1-\varphi)}}{(\varphi - \kappa)^2}. \tag{3.12}$$

In Theorem 3.5 we restrict our analysis to Gaussian sketches and consider $\operatorname{corr}(\mu^2(\boldsymbol{b}, \boldsymbol{S}), \mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}))$ (rather than the more natural quantity $\operatorname{corr}(\mu(\boldsymbol{b}, \boldsymbol{S}), \mu(\tilde{\boldsymbol{b}}, \boldsymbol{S}))$) in order to make analysis tractable. In general $\operatorname{corr}(\mu(\boldsymbol{b}, \boldsymbol{S}), \mu(\tilde{\boldsymbol{b}}, \boldsymbol{S}))$ and $\operatorname{corr}(\mu^2(\boldsymbol{b}, \boldsymbol{S}), \mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}))$ may have significantly different statistical properties. However, if either of them is close to 1, then that would indicate a monotonically increasing (although not necessarily linear) relationship between $\mu(\boldsymbol{b}, \boldsymbol{S})$ and $\mu(\tilde{\boldsymbol{b}}, \boldsymbol{S})$, and when such a relationship holds we expect the boosting procedure in Algorithm 3.1 to work well. While we restrict our attention to Gaussian sketches, this probabilistic model is usually a good indicator of how other sketches perform [32, Remark 8.2]. That is, we expect the result to carry over to the sampling-based sketches (e.g., leverage scores) that we consider. We verify this numerically in section 4.

Remark 3.6. The lower bound in (3.12) is useful only when the right-hand side is close to 1, which roughly requires φ to be large and κ to be small. See Remark 3.9 for how this condition relates to Theorem 3.2.

The rest of this section is organized as follows. Section 3.4 derives some preliminary technical results. Section 3.5 then proves Theorem 3.2. Section 3.6 provides theoretical guarantees for when various sketches satisfy the (ε, δ) pair condition in Definition 2.1 and discuss how this condition in turn ensures that those sketching operators satisfy the requirements in Theorem 3.2. The proof of Theorem 3.5 is given in section SM5 of the supplementary material.

3.4. Preliminary technical results. Our first task is to understand how the optimal residual $r(\boldsymbol{A}, \boldsymbol{b})$ compares to $r_{\boldsymbol{S}}(\boldsymbol{A}, \boldsymbol{b})$. Throughout this section let $\boldsymbol{Q} = \operatorname{orth}(\boldsymbol{A})$.

Lemma 3.7. *Given a sketch matrix $\boldsymbol{S}$, assume* $\ker(\boldsymbol{S}) \cap \operatorname{range}(\boldsymbol{A}) = \{\boldsymbol{0}\}$*, or, equivalently,* $\operatorname{rank}(\boldsymbol{SA}) = \operatorname{rank}(\boldsymbol{A})$*. Then we have*

$$r_{\boldsymbol{S}}^2(\boldsymbol{A}, \boldsymbol{b}) = r^2(\boldsymbol{A}, \boldsymbol{b}) + \|(\boldsymbol{SQ})^\dagger \boldsymbol{SQ}_\perp \boldsymbol{Q}_\perp^T \boldsymbol{b}\|_2^2. \tag{3.13}$$

The proof is contained in subsection SM3.1. We conclude that $r_{\boldsymbol{S}}(\boldsymbol{A}, \boldsymbol{b})$ is comparable to $r(\boldsymbol{A}, \boldsymbol{b})$ if and only if $\|(\boldsymbol{SQ})^\dagger \boldsymbol{SQ}_\perp \boldsymbol{Q}_\perp^T \boldsymbol{b}\|_2^2$ is small, which motivates the definition of μ in (3.5).

Our second result relates the quantities ν, φ, and κ defined in (3.7) and (3.11).

Proposition 3.8. *Assume $\varphi \geq \kappa$. Then we have the two inequalities*

$$\nu \geq \varphi - \kappa \min\left\{1, \sqrt{2(1-\varphi+\kappa)}\right\}, \tag{3.14}$$

$$\nu \geq \varphi - (\varphi\tilde{\kappa} + \sqrt{1-\varphi^2}) \min\left\{1, \sqrt{2(1-\varphi+\varphi\tilde{\kappa}+\sqrt{1-\varphi^2})}\right\}, \tag{3.15}$$

where

$$\tilde{\kappa} \overset{\text{def}}{=} \frac{\|\boldsymbol{P}_{\boldsymbol{Q}} \tilde{\boldsymbol{b}}\|_2}{\|\tilde{\boldsymbol{b}}\|_2} \tag{3.16}$$

measures the relative energy of the low-fidelity vector in the range of $\boldsymbol{A}$.

The proof is provided in subsection SM3.2. The main appeal of (3.15) is that the quantity $\tilde{\kappa}$ involves only low-fidelity data, and hence can be estimated. That is, (3.15) gives a more practically estimable lower bound for ν, involving one quantity $\tilde{\kappa}$ that depends only on low-fidelity data $\tilde{\boldsymbol{b}}$, and the correlation φ between $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$.

Remark 3.9. Recall that our main convergence result, Theorem 3.2, has more attractive bounds when ν is large. By (3.14), ν is large if $\varphi \approx 1$ and $\varphi \gg \kappa$, which coincides with sufficient conditions to ensure attractive bounds in (3.12) in Theorem 3.5 (cf. Remark 3.6). Thus, $\varphi \gg \kappa$ is a unifying condition under which both of our main theoretical results, Theorems 3.2 and 3.5, provide useful bounds. The condition $\varphi \gg \kappa$ means that the correlation between $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ is high and strongly dominates the relative energy of $\boldsymbol{b}$ in range$(\boldsymbol{A})$. Since μ is defined relative to $r(\boldsymbol{A}, \boldsymbol{b})$, a small $r_{\boldsymbol{S}_{\ell^*}}(\boldsymbol{A}, \boldsymbol{b})$ may still result in a large $\mu(\boldsymbol{b}, \boldsymbol{S}_{\ell^*})$ even if $r_{\boldsymbol{S}_{\ell^*}}(\boldsymbol{A}, \boldsymbol{b})$ is small but relatively large compared to $r(\boldsymbol{A}, \boldsymbol{b})$.

3.5. Proof of Theorem 3.2. We first consider the case $\operatorname{corr}(\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b}, \boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}}) \geq 0$. Fixing $\ell \in [L]$, $\boldsymbol{S} = \boldsymbol{S}_\ell$, consider the event E of probability at least $1 - \delta/L$, where the rank condition in (2.4) holds. On this event, this rank condition with Lemma 3.7 implies that

$$r^2_{\boldsymbol{S}}(\boldsymbol{A},\boldsymbol{b}) - r^2(\boldsymbol{A},\boldsymbol{b}) = \|(\boldsymbol{SQ})^\dagger \boldsymbol{SQ}_\perp \boldsymbol{Q}_\perp^T \boldsymbol{b}\|_2^2,$$

allowing us to directly estimate the difference between $\mu(\boldsymbol{b},\boldsymbol{S})$ and $\mu(\tilde{\boldsymbol{b}},\boldsymbol{S})$ as follows:

(3.17)
$$\begin{aligned}
|\mu(\boldsymbol{b},\boldsymbol{S}) - \mu(\tilde{\boldsymbol{b}},\boldsymbol{S})| &= \left| \frac{\|(\boldsymbol{SQ})^\dagger \boldsymbol{SQ}_\perp \boldsymbol{Q}_\perp^T \boldsymbol{b}\|_2}{\|\boldsymbol{Q}_\perp \boldsymbol{Q}_\perp^T \boldsymbol{b}\|_2} - \frac{\|(\boldsymbol{SQ})^\dagger \boldsymbol{SQ}_\perp \boldsymbol{Q}_\perp^T \tilde{\boldsymbol{b}}\|_2}{\|\boldsymbol{Q}_\perp \boldsymbol{Q}_\perp^T \tilde{\boldsymbol{b}}\|_2} \right| \\
&\le \left\| (\boldsymbol{SQ})^\dagger \boldsymbol{S} \left((\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b})_{\mathcal{P}} - (\boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}})_{\mathcal{P}} \right) \right\|_2 \\
&= \|(\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b})_{\mathcal{P}} - (\boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}})_{\mathcal{P}}\|_2 \|(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}\|_2 \\
&= \sqrt{\|(\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b})_{\mathcal{P}}\|_2^2 + \|(\boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}})_{\mathcal{P}}\|_2^2 - 2\langle (\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b})_{\mathcal{P}}, (\boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}})_{\mathcal{P}} \rangle}\, \|(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}\|_2 \\
&= \sqrt{2-2\nu} \cdot \|(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}\|_2 \\
&= \sqrt{2-2\nu} \cdot \|\boldsymbol{Q}(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}\|_2,
\end{aligned}$$

where the first inequality follows from the reverse triangle inequality, the second-to-last equality follows (3.7), and the final equality follows from unitary invariance of the operator norm. The case $\mathrm{corr}(\boldsymbol{P}_{\boldsymbol{Q}_\perp} \boldsymbol{b}, \boldsymbol{P}_{\boldsymbol{Q}_\perp} \tilde{\boldsymbol{b}}) < 0$ can be treated similarly by noting that the inequality on the second line of (3.17) still holds if the minus sign on the right-hand side is changed to a plus sign. The rest of the computation is similar to the case with nonnegative correlation.

Note that $(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}$ is the $\boldsymbol{S}$-sketched least squares solution to $\min_{\boldsymbol{x}} \|\boldsymbol{Qx} - \boldsymbol{h}\|_2$. Also, note that $\boldsymbol{h} \in \mathrm{range}(\boldsymbol{Q}_\perp)$. Using (2.4), the following also holds on the event E:

(3.18)
$$\|\boldsymbol{Q}(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}\|_2^2 + \|\boldsymbol{h}\|_2^2 = \|\boldsymbol{Q}(\boldsymbol{SQ})^\dagger \boldsymbol{Sh} - \boldsymbol{h}\|_2^2 \le (1+\varepsilon)^2 \min_{\boldsymbol{x} \in \mathbb{R}^d} \|\boldsymbol{Qx} - \boldsymbol{h}\|_2^2 = (1+\varepsilon)^2 \|\boldsymbol{h}\|_2^2.$$

Rearranging terms and noting $\|\boldsymbol{h}\|_2 = 1$ yields $\|\boldsymbol{Q}(\boldsymbol{SQ})^\dagger \boldsymbol{Sh}\| \le \sqrt{3\varepsilon}$, which is substituted into (3.17), implying that on an event E with probability at least $1 - \delta/L$, we have

$$|\mu(\boldsymbol{b},\boldsymbol{S}) - \mu(\tilde{\boldsymbol{b}},\boldsymbol{S})| \le \sqrt{6(1-\nu)\varepsilon}. \tag{3.19}$$

Taking a union bound over $\ell \in [L]$ yields that, with probability at least $1 - \delta$,

$$\max_{\ell \in [L]} |\mu(\boldsymbol{b},\boldsymbol{S}_\ell) - \mu(\tilde{\boldsymbol{b}},\boldsymbol{S}_\ell)| \le \sqrt{6(1-\nu)\varepsilon}. \tag{3.20}$$

Conditioning on the event in (3.20) and using the definition of ℓ^* and ℓ^{**} finishes the proof:

$$\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^*}) \le \mu(\tilde{\boldsymbol{b}},\boldsymbol{S}_{\ell^*}) + \sqrt{6(1-\nu)\varepsilon} \le \mu(\tilde{\boldsymbol{b}},\boldsymbol{S}_{\ell^{**}}) + \sqrt{6(1-\nu)\varepsilon} \le \mu(\boldsymbol{b},\boldsymbol{S}_{\ell^{**}}) + 2\sqrt{6(1-\nu)\varepsilon}. \tag{3.21}$$

3.6. Achieving the (ε,δ) pair condition. We next show that, for a variety of random sketches of interest, the $(\varepsilon, \frac{\delta}{L})$ pair condition for $(\boldsymbol{Q},\boldsymbol{h})$ in Theorem 3.2 holds for sufficiently large m. We begin with a lemma that gives a sufficient condition for verification of the $(\varepsilon, \frac{\delta}{L})$ pair condition for $(\boldsymbol{Q},\boldsymbol{h})$, which can be deduced as a special case from [15, Lemma 1].

Lemma 3.10 (see [15]). *Let $\boldsymbol{Q}$ and $\boldsymbol{h}$ be defined as in Theorem 3.2. The distribution of $\boldsymbol{S}$ is an $(\varepsilon, \frac{\delta}{L})$ pair for $(\boldsymbol{Q}, \boldsymbol{h})$ if with probability at least $1-\delta/L$ we simultaneously have*

$$\sigma_{\min}^2(\boldsymbol{S}\boldsymbol{Q}) \geq \frac{\sqrt{2}}{2} \quad \text{and} \quad \|\boldsymbol{Q}^T\boldsymbol{S}^T\boldsymbol{S}\boldsymbol{h}\|_2^2 \leq \frac{\varepsilon}{2}, \tag{3.22}$$

where $\sigma_{\min}(\cdot)$ denotes the smallest singular value of a matrix.

When the conditions in Lemma 3.10 hold, one can directly bound (3.17) using the submultiplicativity of operator norms instead of resorting to an (ε, δ) argument as in the proof of Theorem 3.2, although the latter is more general. Theorem 3.11 presents constructive strategies for generating sketch distributions—based on sub-Gaussian random variables and leverage scores—that achieve appropriate (ε, δ) pair conditions. We recall that a random variable X is called sub-Gaussian if, for some $K > 0$, we have $\mathbb{E}\exp(X^2/K^2) \leq 2$ [39, Definition 2.5.6]. The sub-Gaussian norm of X is defined as $\|X\|_{\psi_2} \stackrel{\text{def}}{=} \inf\left\{K > 0 : \mathbb{E}\exp(X^2/K^2) \leq 2\right\}$ [39]. A proof of Theorem 3.11 is give in section SM6. Variants of these results have appeared previously in the literature [13, 14, 15, 28].

Theorem 3.11. *Let $\boldsymbol{Q}$ and $\boldsymbol{h}$ be defined as in Theorem 3.2, with $\boldsymbol{Q} = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_d]$ the columns of $\boldsymbol{Q}$. Denote by $q_{ij} \stackrel{\text{def}}{=} \boldsymbol{q}_i(j)$ and $h_j \stackrel{\text{def}}{=} \boldsymbol{h}(j)$ the jth component of $\boldsymbol{q}_i$ and $\boldsymbol{h}$, respectively.*

1. *Suppose $\boldsymbol{S} \in \mathbb{R}^{m\times N}$ is a dense sketch whose entries are i.i.d. sub-Gaussian random variables with mean 0 and variance $1/m$. Assume the sub-Gaussian norm of each entry of $\sqrt{m}\boldsymbol{S}$ is bounded by $K \geq 1$. Then the distribution of $\boldsymbol{S}$ is an $(\varepsilon, \frac{\delta}{L})$ pair for $(\boldsymbol{Q}, \boldsymbol{h})$ if*

$$m \geq \frac{CK^4}{\varepsilon} d \log\left(\frac{4dL}{\delta}\right), \tag{3.23}$$

   where C is an absolute constant.

2. *Suppose $\boldsymbol{S} \in \mathbb{R}^{m\times N}$ is a row sketch based on the leverage scores of $\boldsymbol{A}$, and $0 < \varepsilon, \delta < 1/2$; see (2.6). Then the distribution of $\boldsymbol{S}$ is an $(\varepsilon, \frac{\delta}{L})$ pair for $(\boldsymbol{Q}, \boldsymbol{h})$ if*

$$m \geq \max\left\{35 d \log\left(\frac{4dL}{\delta}\right), \frac{2dL}{\varepsilon\delta}\right\}. \tag{3.24}$$

   Moreover, if

$$\max_{i\in[d]} \max_{j\in[N]:\ell_j>0} \frac{d|q_{ij}h_j|}{\ell_j} \leq C, \quad \ell_j = \sum_{k\in[d]} q_{kj}^2 \tag{3.25}$$

   for some constant $C > 0$, then the distribution of $\boldsymbol{S}$ is an $(\varepsilon, \frac{\delta}{L})$ pair for $(\boldsymbol{Q}, \boldsymbol{h})$ if

$$m \geq \max\left\{35, \frac{4C^2}{\varepsilon}\right\} d \log\left(\frac{4dL}{\delta}\right). \tag{3.26}$$

The scalar ℓ_j in (3.25) is the leverage score associated to row j of $\boldsymbol{A}$, and $(\ell_j)_{j\in[N]}$ defines a (discrete) probability distribution over the row indices $[N]$ of $\boldsymbol{A}$; see (2.6).

Remark 3.12. When $\boldsymbol{Q}$ is *incoherent,* i.e., when its leverage scores satisfy $\ell_i = \mathcal{O}(d/N)$ for $i \in [N]$, then the entries q_{ij} satisfy $q_{ij} = \mathcal{O}(1/\sqrt{N})$. For any $\boldsymbol{h}$ such that $\max_{j\in[N]} |h_j| \lesssim \mathcal{O}(1/\sqrt{N})$, the condition in (3.25) is satisfied with $C = \mathcal{O}(1)$:

$$\max_{i\in[d]}\max_{j\in[N]:\ell_j>0}\frac{d|q_{ij}h_j|}{\ell_j}\lesssim\frac{d\cdot\frac{1}{\sqrt{N}}\cdot\frac{1}{\sqrt{N}}}{\frac{d}{N}}=1.$$

Remark 3.13. As noted in section 2.2.3, leveraged volume sampling requires $m \gtrsim d\log(d/\delta)+d/(\varepsilon\delta)$ samples to satisfy the (ε,δ) pair condition. This result appears in Corollary 10 of [9].

4. Numerical experiments.

In this section we illustrate various aspects of the BFB approach using both the manufactured data in section 4.1 as well as data obtained from PDE solutions in section 4.2. The experiments in section 4.2 focus on the polynomial chaos class of emulators [19, 41], in which we employed both quadrature grids and Monte Carlo random sampling grids for constructing the emulator to illustrate the application scope of our proposed methodology. It is worth noting that our analysis and algorithm hold for any emulator construction falling into the general least squares procedure (1.3).

The codes used to generate the results of this section are available from the GitHub repository `https://github.com/CU-UQ/BF-Boosted-Quadrature-Sampling`.

4.1. Verification of theoretical results on synthetic data.

We first verify the theoretical results in Theorems 3.2 and 3.5. We do this by simulating different values for $\boldsymbol{S}$, $\boldsymbol{b}$, and $\tilde{\boldsymbol{b}}$. We generate a design matrix $\boldsymbol{A}\in\mathbb{R}^{1000\times 50}$ (i.e., $N=1000$ and $d=50$) with i.i.d. standard normal entries and fix it in the rest of the simulations. For sketching matrices $\boldsymbol{S}$, we choose the embedding dimension to be $m=100$ and consider both the Gaussian and leverage score sampling sketches. We generate multiple different versions of the vectors $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ that correspond to different values of κ and φ. Recall that these parameters control how much of $\boldsymbol{b}$ is in the range of $\boldsymbol{A}$ and the absolute value of the correlation between $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$, respectively. The vectors are generated via

$$\boldsymbol{b}=\kappa\boldsymbol{Q}\boldsymbol{z}_1+\sqrt{1-\kappa^2}\boldsymbol{Q}_\perp\boldsymbol{z}_2,\qquad \tilde{\boldsymbol{b}}=\varphi\boldsymbol{b}+\sqrt{1-\varphi^2}\boldsymbol{b}_\perp\boldsymbol{z}_3, \tag{4.1}$$

where $\boldsymbol{Q}=\operatorname{orth}(\boldsymbol{A})$, and $\boldsymbol{z}_1\in\mathbb{R}^{d-1}$, $\boldsymbol{z}_2\in\mathbb{R}^{N-d-1}$, and $\boldsymbol{z}_3\in\mathbb{R}^{N-2}$ are generated by normalizing random vectors of appropriate length whose entries are i.i.d. standard normal. The vectors $\boldsymbol{z}_1,\boldsymbol{z}_2,\boldsymbol{z}_3$ are drawn once and then kept fixed for the different choices of κ and φ.

To check the upper bound in Theorem 3.2, we generate $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ using 9 equispaced values for φ and κ between 0 and 1, corresponding to 81 experiments for each sketching strategy. We use a sequence of $L=10$ independent sketching operators in our BFB approach. After computing values of ν for each experiment, we evaluate the optimality coefficient difference $\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^*})-\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^{**}})$. Figure 1 illustrates the resulting relationship between $\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^*})-\mu(\boldsymbol{b},\boldsymbol{S}_{\ell^{**}})$ and the bound $2\sqrt{6(1-\nu)\varepsilon}$. Due to the unknown constants in (3.23) and (3.24), an exact value of ε corresponding to $m=100$ is unavailable. Instead, we choose ε to be 0.01 heuristically. We chose this particular value of ε since it illustrates how the green curve's shape, which is independent of the scalar ε, separates most of the scatter plots from the rest of the area. The result shows our proposed BFB bound in Theorem 3.2 is effective and nonvacuous for both Gaussian and leverage score sketching. It is noticeable that all the dots out of our proposed bound (green) are leverage score sketch spots (blue). The reason is because we set $m=100$ for both sketch strategies, but leverage score sketch requires a higher m to satisfy the (ε,δ) pair condition, which leads to a higher deviation in μ with fixed m; see details in Theorem 3.11.

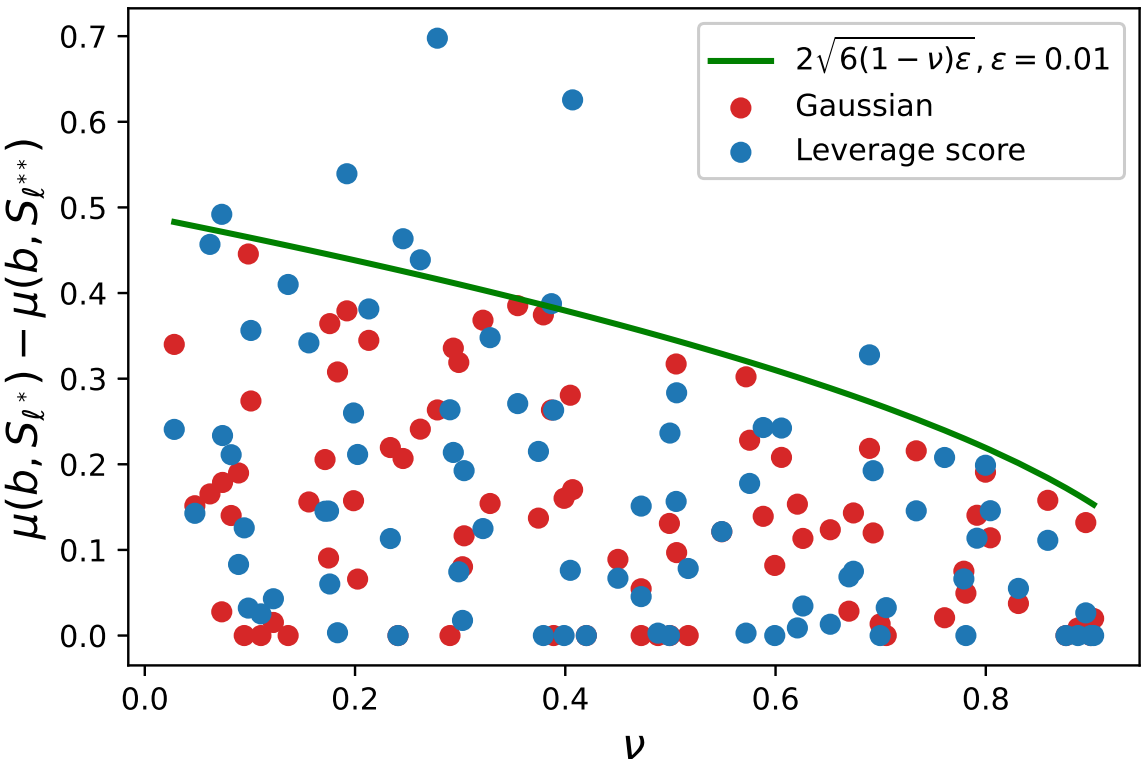

Figure 1. *Scatter plots of $\mu(\boldsymbol{b}, \boldsymbol{S}_{\ell^*}) - \mu(\boldsymbol{b}, \boldsymbol{S}_{\ell^{**}})$ based on given values of ν for Gaussian sketch (red) and leverage score sketch (blue). The green curve is the bound we provide in Theorem 3.2 with $\varepsilon = 0.01$.*

Table 1

Empirical correlation between $\mu^2(\boldsymbol{A}, \boldsymbol{b})$ and $\mu^2(\boldsymbol{A}, \tilde{\boldsymbol{b}})$ for four different parameter setups and two different sketch types.

κ	φ	Sketch type	Correlation
0.2	0.3	Gaussian	0.21
0.2	0.95	Gaussian	0.88
0.95	0.3	Gaussian	0.17
0.95	0.95	Gaussian	0.48
0.2	0.3	Leverage score	0.19
0.2	0.95	Leverage score	0.91
0.95	0.3	Leverage score	0.08
0.95	0.95	Leverage score	0.56

To further validate our theoretical results in Theorem 3.5, we consider four combinations of κ and φ as listed in Table 1, right. For both the Gaussian and leverage score sketches we draw 100 sketches randomly. The same set of sketches is used for each pair of the vectors $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$. Figure 2 shows scatter plots of the squared optimality coefficients for the four different pairs of $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ and two different sketch types.

Table 1 provides the estimated correlations between $\mu^2(\boldsymbol{b}, \boldsymbol{S})$ and $\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S})$ for each of the eight setups based on the data points in Figure 2. For both sketches, a small value of κ and a

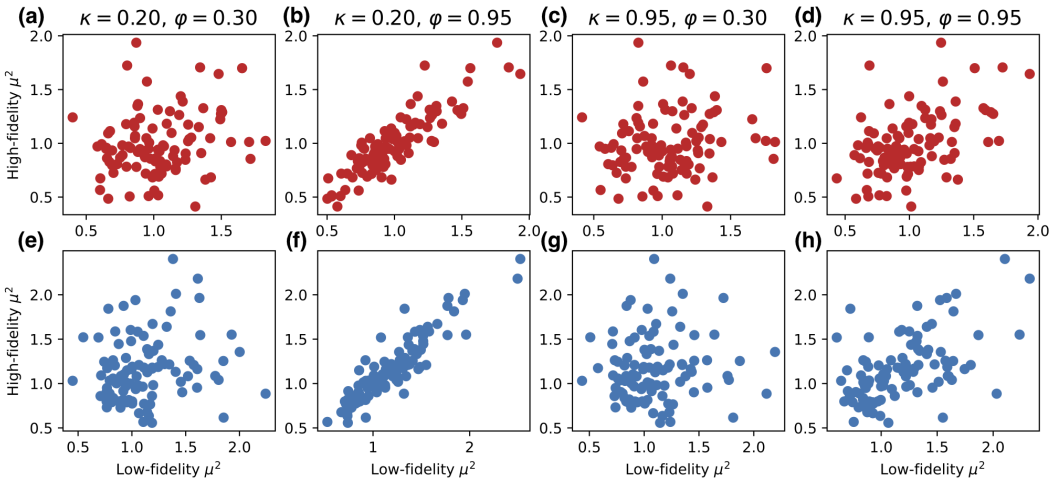

Figure 2. *Scatter plots of the square of the optimality coefficient for high- and low-fidelity data for each of* 100 *different sketches. Each point is equal to* $(\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}), \mu^2(\boldsymbol{b}, \boldsymbol{S}))$ *for one realization of the sketch* $\boldsymbol{S}$*. Gaussian sketches are* (a)–(d) *and leverage score sketches are* (e)–(h).

large value of φ together yield the highest positive correlation between $\mu^2(\boldsymbol{b}, \boldsymbol{S})$ and $\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S})$. In this case, the sketch that attains the smallest residual on the low-fidelity data also attains a near-minimal residual on the high-fidelity data, which is consistent with the upper bound in (3.7) and the lower bound in (3.12) that support the idea of BFB.

4.2. Experiments on PDE datasets. In this section we verify the accuracy of Algorithm 3.1 on two PDE problems: thermally driven cavity fluid flow (section 4.2.1) and simulation of a composite beam (section 4.2.2). In doing so, we consider three sampling strategies based on uniform, leverage score (section 2.2.2), and leveraged volume (section 2.2.3) sampling. As a baseline, we present results from deterministic sketching via column-pivoted QR decomposition (section 2.2.1). We also show the BFB results of random sampling, in which the design matrix $\boldsymbol{A}$ is built with points drawn from the measure induced by the input randomness, instead of a deterministic quadrature rule. The experiments with different design matrices show the BFB method is applicable to a wider range of least square settings.

In both PDE problems, the high-fidelity solution operator takes uniformly distributed inputs $\boldsymbol{p} \in [-1, 1]^q$. We therefore consider approximations of the form in (1.2) with $\psi_j : [-1, 1]^q \mapsto \mathbb{R}$ chosen to be products of q univariate (normalized) Legendre polynomials. Specifically, let $\boldsymbol{j} = (j_1, \ldots, j_q)$, $j_k \in \mathbb{N} \cup \{0\}$, be a vector of nonnegative indices, and let $\psi_{j_k}(p_k)$ denote the Legendre polynomial of degree j_k in p_k such that $\mathbb{E}[\psi^2_{j_k}(p_k)] = 1$. We also need to choose the quadrature nodes $\boldsymbol{p}_n$ and weights w_n, which we choose to be constructed from a tensor-product quadrature. Thus, the entries of $\boldsymbol{A}$ in (1.3) are constructed by defining

$$\psi_{\boldsymbol{j}}(\boldsymbol{p}) = \prod_{k=1}^{q} \psi_{j_k}(p_k), \qquad \boldsymbol{p}_n = (p_{1,n_1}, p_{2,n_2}, \ldots, p_{q,n_q}), \qquad w_n = \prod_{k=1}^{q} w_{k,n_k}. \tag{4.2}$$

There are two spaces with different orders considered in the experiments. The first space, known as the total degree space, involves the selection of each column $\boldsymbol{j}$ based on whether

the condition $\sum_{k=1}^{q} j_k \le \zeta$ holds, where $\zeta \ge 0$ is a fixed value. The second space we examine is referred to as the hyperbolic cross space [16], in which the columns $\boldsymbol{j}$ must satisfy the condition $\prod_{k=1}^{q}(j_k+1) \le \zeta+1$. It's worth noting that the order of the hyperbolic cross space is smaller than that of the total degree space. This contrast provides a basis for comparing the performance of spaces with differing degrees.

Each sequence $(p_{k,n_k}, w_{k,n_k})_{n_k\in[N_k]}$ consists of node-weight pairs in the N_k-point Gauss–Legendre quadrature on $[-1,1]$. The resulting sequence $(\boldsymbol{p}_n, w_n)_{n\in[N]}$ contains $N = \prod_{k=1}^{q} N_k$ pairs. When $\boldsymbol{A}$ is constructed in this fashion, it is possible to sample rows of that matrix according to the exact leverage score using the efficient method by [31]; see section SM2. As an aside, we remark that such tensorial constructions of one-dimensional quadrature rules have accuracy guarantees for smooth functions that decrease exponentially in the number of points *per dimension,* e.g., [4, Theorem 3.2] and [2, section 3]. This construction, unfortunately, suffers from the curse of dimensionality, which motivates why one might prefer compression/sketching strategies as we propose here. As an alternative to tensor-product constructions, sparse grids could also be used [3] to identify a quadrature-type sampling for least squares problems that have error guarantees. However, the consideration of how one identifies a "good" full least squares problem is not the goal of this article; instead our focus is on accurately approximating the full least squares solution when the corresponding data collection is computationally unaffordable.

With $\hat{\boldsymbol{x}}_{\mathsf{BFB}}$ the output from Algorithm 3.1, we measure performance via the relative error,

$$E \overset{\text{def}}{=} \frac{\|\boldsymbol{A}\hat{\boldsymbol{x}}_{\mathsf{BFB}} - \boldsymbol{b}\|_2}{\|\boldsymbol{b}\|_2}. \tag{4.3}$$

4.2.1. Cavity fluid flow. Here we consider the case of temperature-driven fluid flow in a two-dimensional cavity [3, 36, 25, 24, 17, 26], with the quantity of interest being the heat flux averaged along the hot wall, as Figure 3 shows. The wall on the left-hand side is the hot wall with random temperature T_h, and the cold wall at the right-hand side has temperature $T_c < T_h$. $\bar{T}_c$ is the constant mean of T_c. The horizontal walls are adiabatic. The reference temperature and the temperature difference are given by $T_{\text{ref}} = (T_h + \bar{T}_c)/2$ and $\Delta T_{\text{ref}} = T_h - \bar{T}_c$, respectively. The normalized governing equations are given by

$$\begin{aligned} &\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \frac{\text{Pr}}{\sqrt{\text{Ra}}}\nabla^2\mathbf{u} + \text{Pr}\Theta\mathbf{e}_y, \quad \frac{\partial \Theta}{\partial t} + \nabla\cdot(\mathbf{u}\Theta) = \frac{1}{\sqrt{\text{Ra}}}\nabla^2\Theta, \\ &\nabla\cdot\mathbf{u} = 0, \qquad\qquad T(x=1,y) = \bar{T}_c + \sigma_T\sum_{i=1}^{q}\sqrt{\lambda_i}\phi_i(y)\mu_i, \end{aligned} \tag{4.4}$$

where $\mathbf{e}_y$ is the unit vector $(0,1)$, $\mathbf{u} = (u,v)$ is the velocity vector field, $\Theta = (T - T_{\text{ref}})/\Delta T_{\text{ref}}$ is normalized temperature, p is pressure, and t is time. We assume no-slip boundary conditions on the walls. The dimensionless Prandtl and Rayleigh numbers are defined as $\text{Pr} = \nu_{\text{visc}}/\alpha$ and $\text{Ra} = g\tau\Delta T_{\text{ref}} W^3/(\nu_{\text{visc}}\alpha)$, respectively, where W is the width of the cavity, g is gravitational acceleration, ν_{visc} is kinematic viscosity, α is thermal diffusivity, and τ is the coefficient of thermal expansion. We set $g = 10$, $W = 1$, $\tau = 0.5$, $\Delta T_{\text{ref}} = 100$, $\text{Ra} = 10^6$, and $\text{Pr} = 0.71$.

On the cold wall, we apply a temperature distribution with stochastic fluctuations through T shown in (4.4), where $\bar{T}_c = 100$ is a constant, $\{\lambda_i\}_{i\in[q]}$ and $\{\phi_i(y)\}_{i\in[q]}$ are the q largest

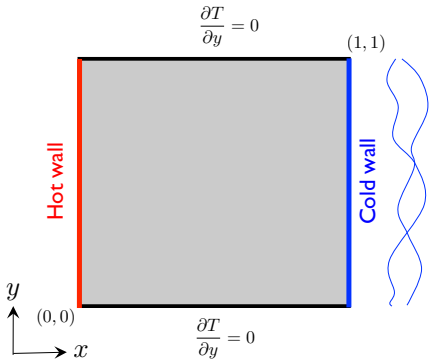

Figure 3. *Schematic of the temperature-driven cavity flow problem, reproduced from Figure 5 of [17].*

Table 2

Correlation coefficients between $\mu^2(\boldsymbol{A}, \boldsymbol{b})$ and $\mu^2(\boldsymbol{A}, \tilde{\boldsymbol{b}})$ for different sampling methods under total degree or hyperbolic cross space. The correlation is computed based on the points shown in Figure 4.

Polynomial space	Uniform sampling	Leverage score sampling	Leveraged volume sampling
Total degree	0.66	0.57	0.18
Hyperbolic cross	0.99	0.98	0.98

eigenvalues and corresponding eigenfunctions of the kernel $k(y_1, y_2) = \exp(-|y_1 - y_2|/0.15)$, and each $\mu_i \overset{\text{i.i.d.}}{\sim} U[-1, 1]$. We let $q = 2$ (though in general, this does not need to match the physical dimension) and $\sigma_T = 2$. The vector $\boldsymbol{p} = (\mu_1, \mu_2)$ is the uncertain input of the model.

In order to solve (4.4) we use the finite volume method with two different grid resolutions: a finer grid of size 128×128 to produce the high-fidelity solution and a coarser grid of size 16×16 to produce the low-fidelity solution. The computational cost ratio between the high-fidelity model and the low-fidelity model is approximately 955.43, as reported from [17]. For our surrogate model, we choose the basis set $\{\psi_j\}_{j \in [d]}$ based on the total degree and hyperbolic cross spaces of maximum order $\zeta = 4$. The corresponding spaces have $d = 15$ and $d = 10$ basis functions, respectively. The quadrature pairs $(\boldsymbol{p}_n, w_n)$ used to construct $\boldsymbol{A}$, $\boldsymbol{b}$, and $\tilde{\boldsymbol{b}}$ are defined as in (4.2) and are based on the nodes and weights from a 40-point Gauss–Legendre rule, i.e., $N_1 = N_2 = 40$.

We first repeat the test we ran on synthetic data in section 4.1. Figure 4 shows the scatter plots of $(\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}), \mu^2(\boldsymbol{b}, \boldsymbol{S}))$ for the two different polynomial spaces and three different quadrature sampling approaches. Each plot is based on 100 sketches with $m = 30$ and $m = 20$ samples used for the total degree and hyperbolic cross spaces, respectively. Table 2 presents the correlation coefficients between $\mu^2(\boldsymbol{b}, \boldsymbol{S})$ and $\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S})$ based on the points in Figure 4. There is a discrepancy between the correlation observed for the total degree and hyperbolic cross spaces. One possible explanation for this is that a greater portion of $\boldsymbol{b}$ is in the range of $\boldsymbol{A}$ for the total degree space than for the hyperbolic cross space, i.e., κ (see (3.11)) is larger for the former space, which by Theorem 3.5 suggests lower correlation.

Next, we run Algorithm 3.1 with $L = 10$ sketches and the number of samples $m = 1.2d$ and $m = 2d$. Figure 5 shows the relative error E in (4.3) from running the algorithm 1000

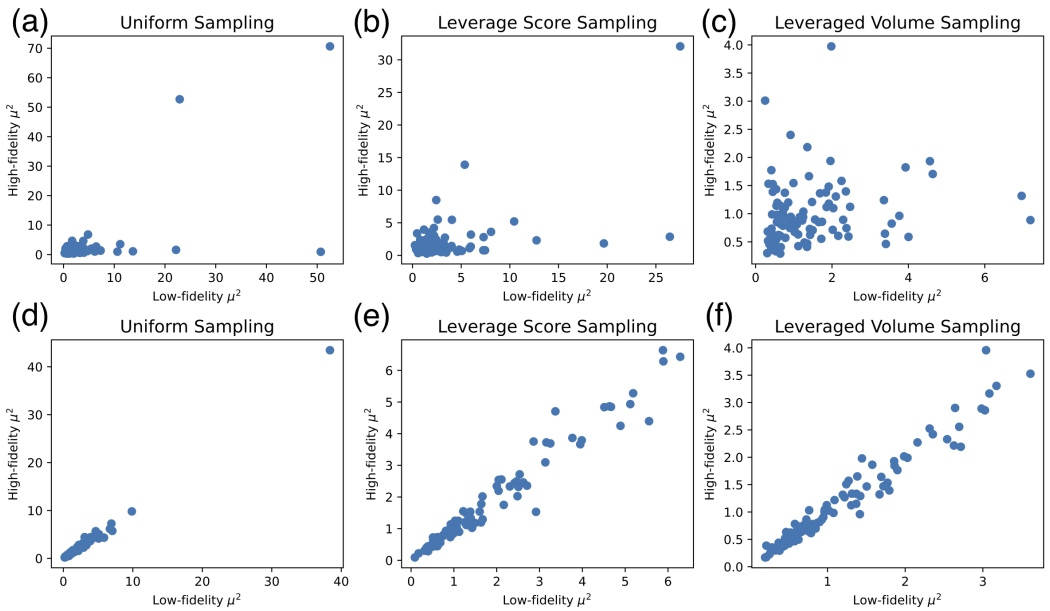

Figure 4. *Scatter plots of the square of the optimality coefficient for high- and low-fidelity data from the cavity fluid flow problem for different polynomial spaces (total degree:* (a)–(c)*; hyperbolic cross:* (d)–(f)*) and types of sampling. Each point is equal to* $(\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}), \mu^2(\boldsymbol{b}, \boldsymbol{S}))$ *for one realization of the sketch* $\boldsymbol{S}$*, and each subplot contains* 100 *points (i.e., is based on* 100 *sketch realizations). For the total degree space,* $m = 30$ *samples are used and for the hyperbolic cross space* $m = 20$ *samples are used.*

times for each of the different choices of polynomial space, sketch size m, and quadrature sampling approach. We observe that in all cases the BFB approach improves the error as compared to the nonboosted case. In particular, the improvement is more considerable in the case of the hyperbolic cross basis, which is explained by the higher correlation between $\mu^2(\boldsymbol{A}, \boldsymbol{b})$ and $\mu^2(\boldsymbol{A}, \tilde{\boldsymbol{b}})$, as reported in Figure 4. Additionally, for the case of hyperbolic space, the BFB results is comparable or better performance as compared to the column-pivoted QR decomposition (blue line in Figure 5). Note that the computational cost of column-pivoted QR is higher than the BFB as it requires the QR decomposition of the entire matrix $\boldsymbol{A}$. Besides the quadrature sampling results in Figure 5, the error E of random sampling is presented in Figure 6. We observe that BFB also improves the performance, which discloses a wider scope of application for the BFB algorithm.

4.2.2. Composite beam. Following [27, 5, 6], we consider a plane-stress, cantilever beam with composite cross section and hollow web as shown in Figure 7. The quantity of interest in this case is the maximum displacement of the top cord. The uncertain parameters of the model are E_1, E_2, E_3, f, where E_1, E_2, and E_3 are the Young's moduli of the three components of the cross section and f is the intensity of the applied distributed force on the beam; see Figure 7. These are assumed to be statistically independent and uniformly distributed. The dimension of the input parameter is therefore $q = 4$. Table 3 shows the range of the input parameters as well as the other deterministic parameters.

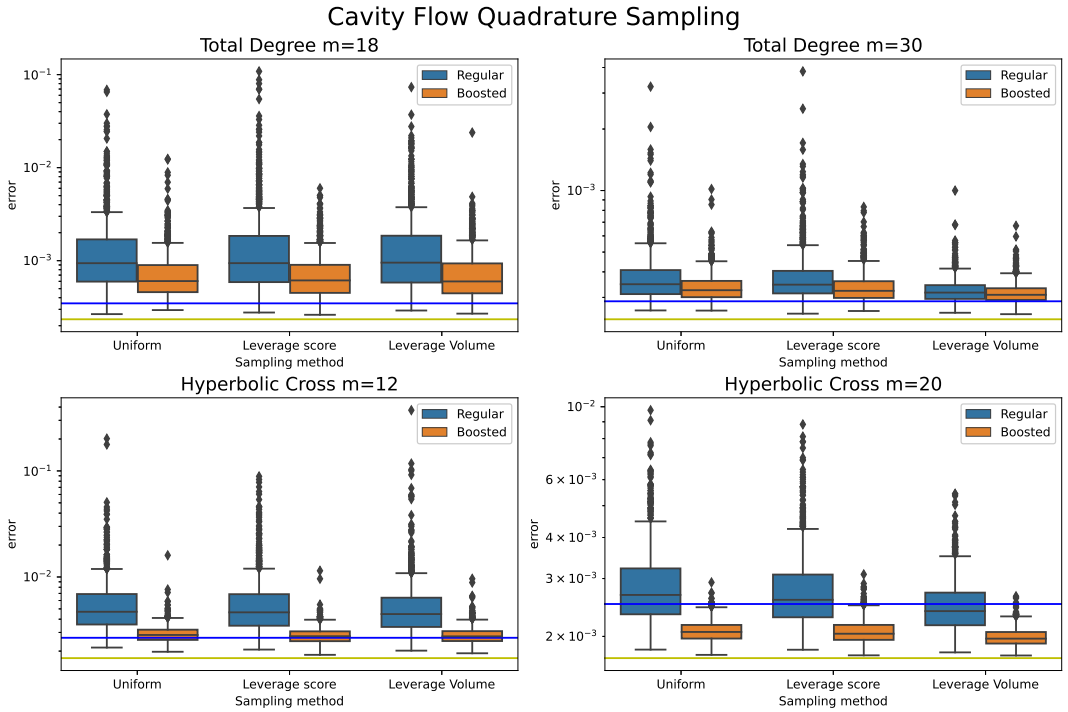

Figure 5. *Relative error for different sampling methods and polynomial spaces when fitting the surrogate model to the cavity fluid flow data. Yellow lines show the relative error E in (4.3) for the unsketched solution in (1.3). Blue lines show E when the coefficients $\boldsymbol{x}$ are computed via the QR decomposition–based method in section 2.2.1. The blue box plots show the distribution of E based on* 1000 *trials when $\boldsymbol{x}$ is computed as in (2.2). The orange box plots show the same thing, but for the solution $\hat{\boldsymbol{x}}_{\mathsf{BFB}}$ computed via Algorithm 3.1.*

For the cavity fluid flow problem in section 4.2.1, we created high- and low-fidelity solutions by changing the resolution of the grid used in the numerical solver. For the present problem, we instead use two different models. The high-fidelity model is based on a finite element discretization on a triangle mesh, as Figure 7 (bottom) shows. The low-fidelity model is derived from Euler–Bernoulli beam theory in which the vertical cross sections are assumed to remain planes throughout the deformation; in particular both the shear deformation of the web and the circular holes are ignored. The Euler–Bernoulli theorem furnishes a differential equation for the vertical displacement u that can be explicitly solved:

$$EI\frac{d^4u(x)}{dx^4} = -f \quad \Longrightarrow \quad u(x) = -\frac{fH^4}{24EI}\left(\left(\frac{x}{H}\right)^4 - 4\left(\frac{x}{H}\right)^3 + 6\left(\frac{x}{H}\right)^2\right), \tag{4.5}$$

where E and I are, respectively, the Young's modulus and the moment of inertia of an equivalent cross section consisting of a single material, and we have taken $E = E_3$, and the width of the top and bottom sections are $w_1 = (E_1/E_3)w$ and $w_2 = (E_2/E_3)w$, while all other dimensions are the same, as Figure 7 shows. Since the low-fidelity data is directly generated from an explicit formula, its computational cost is negligible.

The quadrature surrogate model is based on multivariate Legendre polynomials of maximum degree $\zeta = 2$ with total degree and hyperbolic cross truncation. The corresponding

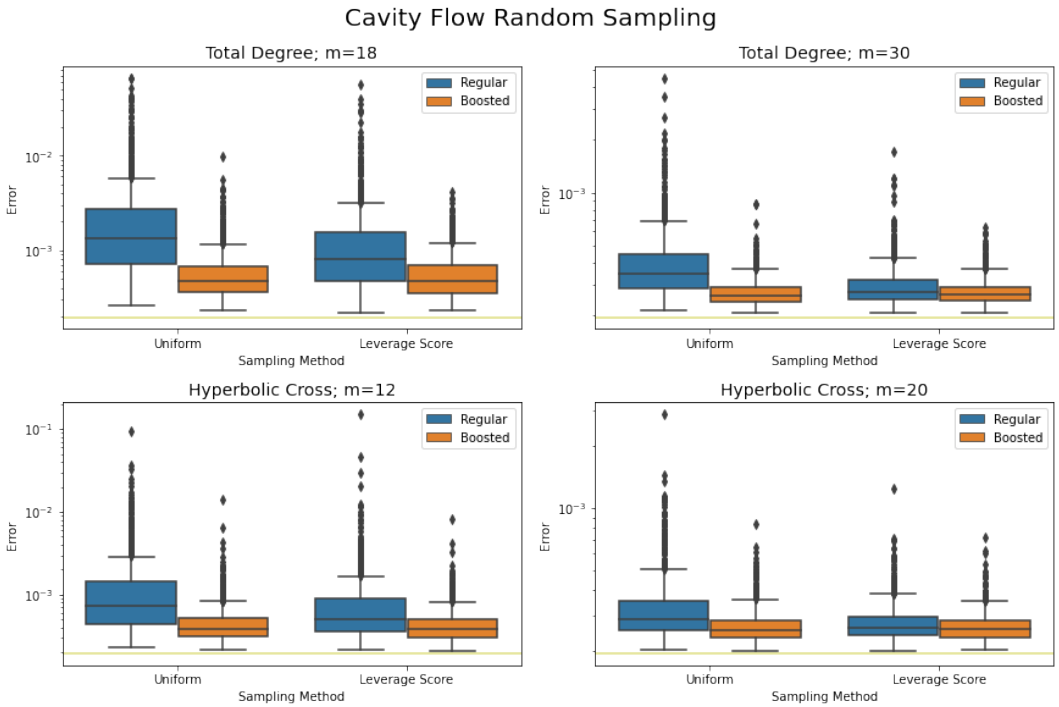

Figure 6. *Same as Figure 5 except that the design matrix $\boldsymbol{A}$ is built with randomly sampled points instead of quadrature grid points.*

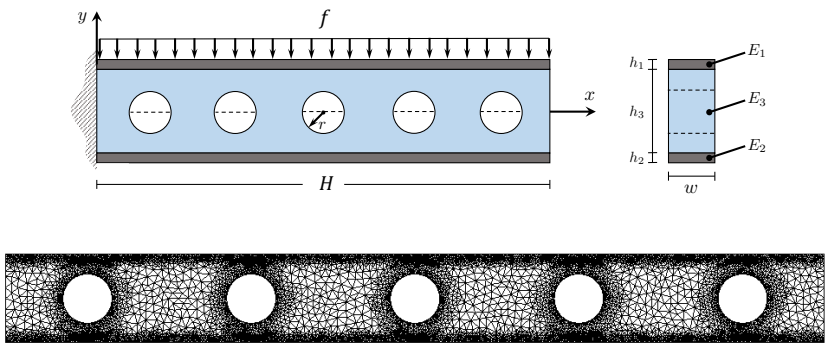

Figure 7. *Top: Cantilever beam (left) and the composite cross section (right) adapted from [27]. Bottom: Finite element mesh used to generate high-fidelity solutions.*

Table 3

The values of the parameters in the composite cantilever beam model. The center of the holes are at $x = \{5, 15, 25, 35, 45\}$. The parameters f, E_1, E_2, and E_3 are drawn independently and uniformly at random from the specified intervals.

H	h_1	h_2	h_3	w	r	f	E_1	E_2	E_3
50	0.1	0.1	5	1	1.5	$[9, 11]$	[0.9e6, 1.1e6]	[0.9e6, 1.1e6]	[0.9e4, 1.1e4]

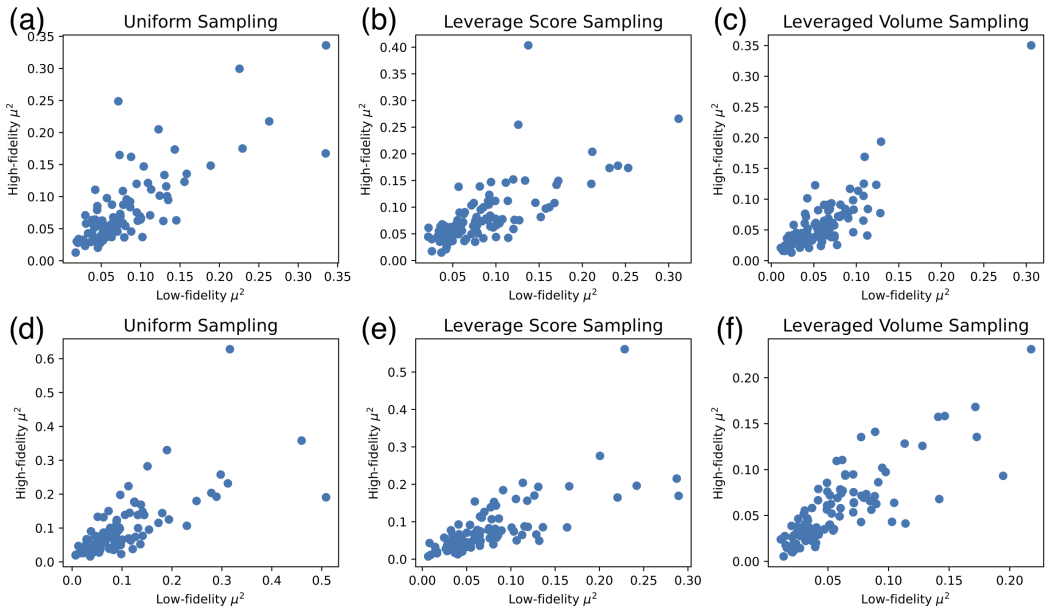

Figure 8. *Scatter plots of the square of the optimality coefficient for high- and low-fidelity data from the composite beam problem for different polynomial spaces (total degree:* (a)–(c)*; hyperbolic cross:* (d)–(f)*) and types of sampling. Each point is equal to* $(\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}), \mu^2(\boldsymbol{b}, \boldsymbol{S}))$ *for one realization of the sketch* $\boldsymbol{S}$*, and each subplot contains* 100 *points (i.e., is based on* 100 *sketch realizations). For the total degree space* $m = 30$ *samples are used, and for the hyperbolic cross space* $m = 18$ *samples are used.*

Table 4

Correlation coefficient between $\mu^2(\boldsymbol{A}, \boldsymbol{b})$ *and* $\mu^2(\boldsymbol{A}, \tilde{\boldsymbol{b}})$ *for different sampling methods under total degree or hyperbolic cross space. The correlation is computed based on the points shown in Figure 8.*

Polynomial space	Uniform sampling	Leverage score sampling	Leveraged volume sampling
Total degree	0.77	0.69	0.84
Hyperbolic cross	0.72	0.73	0.82

spaces have $d = 15$ and $d = 9$ basis functions, respectively. As in the case of the cavity flow problem, the quadrature pairs $(\boldsymbol{p}_n, w_n)$ used to construct $\boldsymbol{A}$, $\boldsymbol{b}$, and $\tilde{\boldsymbol{b}}$ are based on the nodes and weights from 10-point Gauss–Legendre rule appropriately mapped into the ranges given in Table 3.

Figure 8 shows the scatter plots of $(\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}), \mu^2(\boldsymbol{b}, \boldsymbol{S}))$ when repeating the experiment in section 4.1 for the different polynomial spaces and quadrature sampling approaches. Each plot is based on 100 sketches with $m = 2d$, i.e., $m = 30$ and $m = 18$ samples used for the total degree and hyperbolic cross spaces, respectively. Table 4 reports the correlation coefficient between $\mu^2(\boldsymbol{b}, \boldsymbol{S})$ and $\mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S})$, indicating an overall high correlation in all cases.

Next, we run Algorithm 3.1 with $L = 10$ sketches and m chosen to be $m = 1.2d$ and $m = 2d$. Figure 9 shows the results from running the algorithm 1000 times for each of the different choices of polynomial space, number of samples m, and quadrature sampling approach. Figure 10 presents the results from randomly sampled grid points under the same

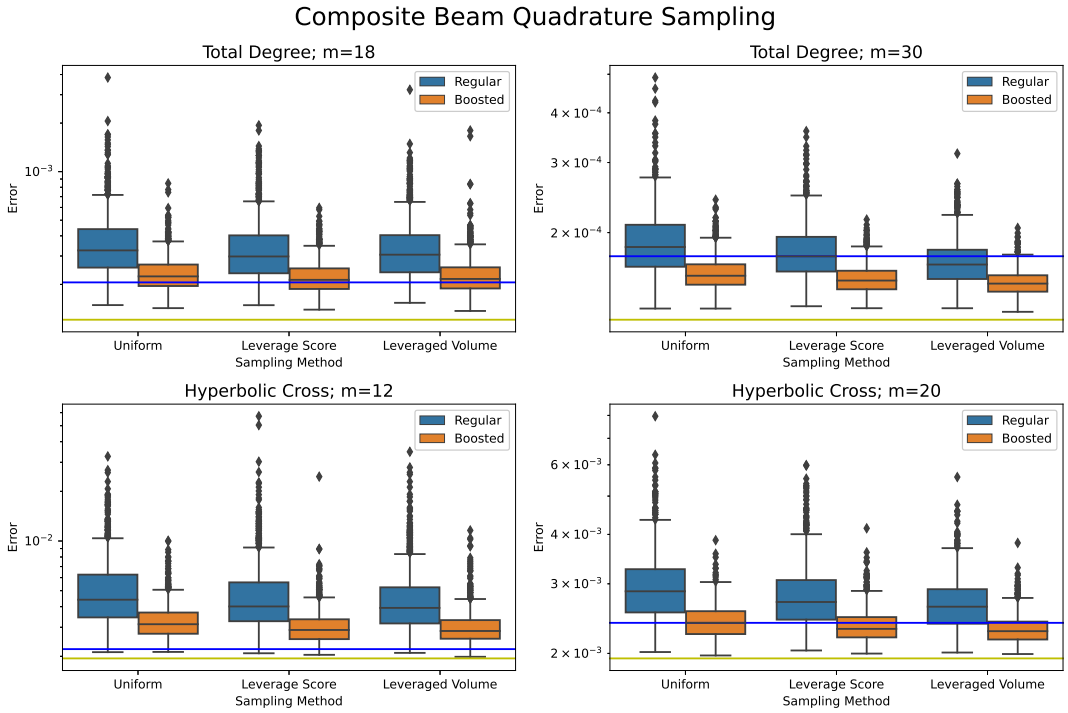

Figure 9. *Relative error for different sampling methods and polynomial spaces when fitting the surrogate model to the beam problem data. Yellow lines show the relative error E in (4.3) for the unsketched solution in (1.3). Blue lines show E when the coefficients $\boldsymbol{x}$ are computed via the QR decomposition–based method in section 2.2.1. The blue box plots shows the distribution of E based on 1000 trials when $\boldsymbol{x}$ is computed as in (2.2). The orange box plots shows the same things, but for the solution $\hat{\boldsymbol{x}}_{\text{BFB}}$ computed via Algorithm 3.1.*

settings. We observe that the BFB performance is superior to that of the nonboosted implementation as it leads to a smaller variance of the error and fewer outliers with smaller deviations from the mean performance for both cases. In the quadrature sampling example, the BFB leads to accuracy comparable to the column-pivoted QR sketch, but with smaller sketching cost. As in the case of the cavity flow, the results corroborate the discussion below Theorem 3.5, in that the BFB improves the regression accuracy when $\text{corr}(\mu^2(\boldsymbol{b}, \boldsymbol{S}), \mu^2(\tilde{\boldsymbol{b}}, \boldsymbol{S}))$ is large.

5. Conclusion.

This work was concerned with the construction of (polynomial) emulators of parameter-to-solution maps of PDE problems via sketched least squares regression. Sketching is a design of experiments approach that aims to improve the cost of building a least squares solution in terms of reducing the number of samples needed—when the cost of generating data is high—or the cost of generating a least squares solution—when data size is substantial. Focusing on the former case, we have proposed a new boosting algorithm to compute a sketched least squares solution.

The procedure consisted in identifying the best sketch from a set of candidates used to construct least squares regression of the low-fidelity data and applying this *optimal* sketch to the regression of high-fidelity data. The bifidelity boosting (BFB) approach limits the required

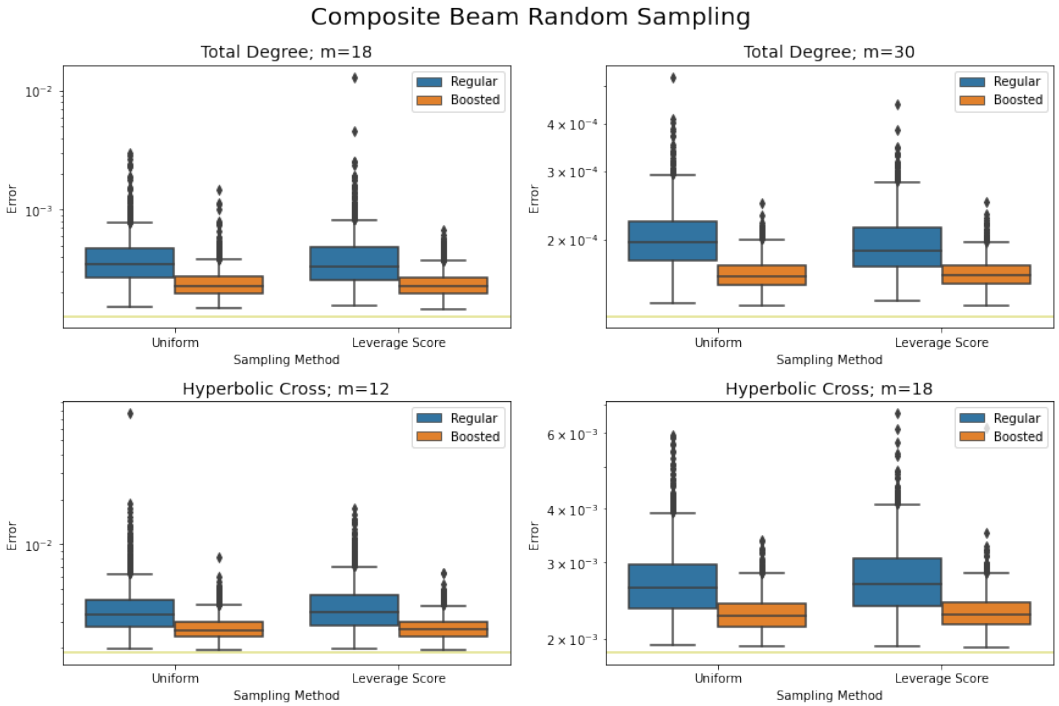

Figure 10. *Same as Figure 9 except that the design matrix* $\boldsymbol{A}$ *is built with randomly sampled points instead of fixed quadrature grid points.*

sample complexity to $\sim d \log d$ high-fidelity data, where d is the size of the (polynomial) basis. We have provided theoretical analysis of the BFB approach identifying assumptions on the low- and high-fidelity data under which the BFB leads to improvement of the solution relative to nonboosted regression of the high-fidelity data. We have also provided quantitative bounds on the residual of the BFB solution relative to the full, computationally expensive solution. We have investigated the performance of BFB on manufactured and PDE data from fluid and solid mechanics. These cover sketching strategies based on leverage score and leveraged volume sampling, for truncated Legendre polynomials of both total degree and hyperbolic cross type. All tests illustrated the efficacy of BFB in reducing the residual—as compared to the nonboosted implementation—and validate the theoretical results.

The present study was focused on the case of (weighted) least squares polynomial regression. When the regression coefficients are sparse, methods based on compressive sampling have proven efficient in reducing the sample complexity below the size of the polynomial basis; see, e.g., [12, 1]. Furthermore, our analysis in this work is built on the assumption that the observations $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ are noise-free. From a more practical perspective, it is more helpful to consider the noise in our analysis. Therefore, some interesting future research directions could include extending the BFB strategy to such underdetermined cases, for instance, using the approach of [10], and building our analysis from the joint randomness from both sketching operators and noisy observations.

REFERENCES

[1] B. Adcock, S. Brugiapaglia, and C. G. Webster, *Sparse Polynomial Approximation of High-Dimensional Functions*, Comput. Sci Engrg. 25, SIAM, 2022, https://doi.org/10.1137/1.9781611976885.

[2] M. Bachmayr and A. Cohen, *Kolmogorov widths and low-rank approximations of parametric elliptic PDEs*, Math. Comp., 86 (2017), pp. 701–724, https://doi.org/10.1090/mcom/3132.

[3] H.-J. Bungartz and M. Griebel, *Sparse grids*, Acta Numer., 13 (2004), pp. 147–269, https://doi.org/10.1017/S0962492904000182.

[4] C. Canuto and A. Quarteroni, *Approximation results for orthogonal polynomials in Sobolev spaces*, Math. Comp., 38 (1982), pp. 67–86, https://doi.org/10.2307/2007465.

[5] S. De, J. Britton, M. Reynolds, R. Skinner, K. Jansen, and A. Doostan, *On transfer learning of neural networks using bi-fidelity data for uncertainty propagation*, Int. J. Uncertain. Quantif., 10 (2020), pp. 543–573.

[6] S. De and A. Doostan, *Neural network training using ℓ_1-regularization and bi-fidelity data*, J. Comput. Phys., 458 (2022), 111010.

[7] M. Dereziński and M. K. Warmuth, *Unbiased estimates for linear regression via volume sampling*, in Advances in Neural Information Processing Systems 30 (NIPS 2017), 2017.

[8] M. Dereziński and M. K. Warmuth, *Reverse iterative volume sampling for linear regression*, J. Mach. Learn. Res., 19 (2018), pp. 853–891.

[9] M. Dereziński, M. K. Warmuth, and D. Hsu, *Leveraged volume sampling for linear regression*, in Advances in Neural Information Processing Systems 31 (NeurIPS 2018), 2018.

[10] P. Diaz, A. Doostan, and J. Hampton, *Sparse polynomial chaos expansions via compressed sensing and D-optimal design*, Comput. Methods Appl. Mech. Engrg., 336 (2018), pp. 640–666.

[11] A. Doostan, R. G. Ghanem, and J. Red-Horse, *Stochastic model reduction for chaos representations*, Comput. Methods Appl. Mech. Engrg., 196 (2007), pp. 3951–3966.

[12] A. Doostan and H. Owhadi, *A non-adapted sparse approximation of PDEs with stochastic inputs*, J. Comput. Phys., 230 (2011), pp. 3015–3034.

[13] P. Drineas, M. W. Mahoney, and S. Muthukrishnan, *Sampling algorithms for ℓ_2 regression and applications*, in Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, 2006, pp. 1127–1136.

[14] P. Drineas, M. W. Mahoney, and S. Muthukrishnan, *Relative-error CUR matrix decompositions*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 844–881, https://doi.org/10.1137/07070471X.

[15] P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlós, *Faster least squares approximation*, Numer. Math., 117 (2011), pp. 219–249.

[16] D. Dung and M. Griebel, *Hyperbolic cross approximation in infinite dimensions*, J. Complexity, 33 (2016), pp. 55–88.

[17] H. R. Fairbanks, A. Doostan, C. Ketelsen, and G. Iaccarino, *A low-rank control variate for multilevel Monte Carlo simulation of high-dimensional uncertain systems*, J. Comput. Phys., 341 (2017), pp. 121–139.

[18] H. R. Fairbanks, L. Jofre, G. Geraci, G. Iaccarino, and A. Doostan, *Bi-fidelity approximation for uncertainty quantification and sensitivity analysis of irradiated particle-laden turbulence*, J. Comput. Phys., 402 (2020), 108996.

[19] R. G. Ghanem and P. D. Spanos, *Stochastic Finite Elements: A Spectral Approach*, Courier Corporation, 2003.

[20] L. Guo, A. Narayan, L. Yan, and T. Zhou, *Weighted approximate Fekete points: Sampling for least-squares polynomial approximation*, SIAM J. Sci. Comput., 40 (2018), pp. A366–A387, https://doi.org/10.1137/17M1140960.

[21] C. Haberstich, A. Nouy, and G. Perrin, *Boosted optimal weighted least-squares*, Math. Comp., 91 (2022), pp. 1281–1315.

[22] M. Hadigol and A. Doostan, *Least squares polynomial chaos expansion: A review of sampling strategies*, Comput. Methods Appl. Mech. Engrg., 332 (2018), pp. 382–407.

[23] N. Halko, P.-G. Martinsson, and J. A. Tropp, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217–288, https://doi.org/10.1137/090771806.
[24] J. Hampton and A. Doostan, *Coherence motivated sampling and convergence analysis of least squares polynomial chaos regression*, Comput. Methods Appl. Mech. Engrg., 290 (2015), pp. 73–97.
[25] J. Hampton and A. Doostan, *Compressive sampling of polynomial chaos expansions: Convergence analysis and sampling strategies*, J. Comput. Phys., 280 (2015), pp. 363–386.
[26] J. Hampton and A. Doostan, *Basis adaptive sample efficient polynomial chaos (BASE-PC)*, J. Comput. Phys., 371 (2018), pp. 20–49.
[27] J. Hampton, H. R. Fairbanks, A. Narayan, and A. Doostan, *Practical error bounds for a non-intrusive bi-fidelity approach to parametric/stochastic model reduction*, J. Comput. Phys., 368 (2018), pp. 315–332.
[28] B. W. Larsen and T. G. Kolda, *Practical leverage-based sampling for low-rank tensor decomposition*, SIAM J. Matrix Anal. Appl., 43 (2022), pp. 1488–1517, https://doi.org/10.1137/21M1441754.
[29] O. Le Maître and O. M. Knio, *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*, Springer Science & Business Media, 2010.
[30] M. W. Mahoney, *Randomized algorithms for matrices and data*, Found. Trends Mach. Learn., 3 (2011), pp. 123–224.
[31] O. A. Malik, Y. Xu, N. Cheng, S. Becker, A. Doostan, and A. Narayan, *Fast Algorithms for Monotone Lower Subsets of Kronecker Least Squares Problems*, preprint, arXiv:2209.05662, 2022.
[32] P.-G. Martinsson and J. A. Tropp, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numer., 29 (2020), pp. 403–572.
[33] A. Narayan, C. Gittelson, and D. Xiu, *A stochastic collocation algorithm with multifidelity models*, SIAM J. Sci. Comput., 36 (2014), pp. A495–A521, https://doi.org/10.1137/130929461.
[34] F. Newberry, J. Hampton, K. Jansen, and A. Doostan, *Bi-fidelity reduced polynomial chaos expansion for uncertainty quantification*, Comput. Mech., 69 (2022), pp. 405–424.
[35] B. Peherstorfer, K. Willcox, and M. Gunzburger, *Survey of multifidelity methods in uncertainty propagation, inference, and optimization*, SIAM Rev., 60 (2018), pp. 550–591, https://doi.org/10.1137/16M1082469.
[36] J. Peng, J. Hampton, and A. Doostan, *A weighted ℓ_1-minimization approach for sparse polynomial chaos expansions*, J. Comput. Phys., 267 (2014), pp. 92–111.
[37] P. Seshadri, A. Narayan, and S. Mahadevan, *Effectively subsampled quadratures for least squares polynomial approximations*, SIAM/ASA J. Uncertain. Quantif., 5 (2017), pp. 1003–1023, https://doi.org/10.1137/16M1057668.
[38] R. C. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications*, Comput. Sci. Engrg. 12, SIAM, 2013, https://doi.org/10.1137/1.9781611973228.
[39] R. Vershynin, *High-Dimensional Probability: An Introduction with Applications in Data Science*, Cambridge University Press, 2018.
[40] D. P. Woodruff, *Sketching as a tool for numerical linear algebra*, Found. Trends Theor. Comput. Sci., 10 (2014), pp. 1–157.
[41] D. Xiu and G. E. Karniadakis, *The Wiener–Askey polynomial chaos for stochastic differential equations*, SIAM J. Sci. Comput., 24 (2002), pp. 619–644, https://doi.org/10.1137/S1064827501387826.
[42] X. Zhu, A. Narayan, and D. Xiu, *Computational aspects of stochastic collocation with multifidelity models*, SIAM/ASA J. Uncertain. Quantif., 2 (2014), pp. 444–463, https://doi.org/10.1137/130949154.