This article has been accepted for publication in Automatica, the final version is available at
https://doi.org/10.1016/j.automatica.2022.110599

# Bayesian frequentist bounds for machine learning and system identification

Giacomo Baggio [a], Algo Carè [b], Anna Scampicchio [c], Gianluigi Pillonetto [a]

[a] *Department of Information Engineering, University of Padova, Padova, Italy*

[b] *Department of Information Engineering, University of Brescia, Brescia, Italy*

[c] *Institute for Dynamic Systems and Control, ETH Zürich, Zürich, Switzerland*

**Abstract**

Estimating a function from noisy measurements is a crucial problem in statistics and engineering, with an impact on machine learning predictions and identification of dynamical systems. In view of robust control design and safety-critical applications such as autonomous driving and smart healthcare, estimates are required to be complemented with uncertainty bounds quantifying their reliability. Most of the available results are derived by constraining the estimates to belong to a deterministic function space; however, the returned bounds often result overly conservative and, hence, of limited usefulness. An alternative is to use a Bayesian framework. The regions thereby obtained however require complete specification of prior distributions whose choice may significantly affect the probability of inclusion. This study presents a framework for the effective computation of regions that include the unknown function with exact probability. In this setting, the users not only have the freedom to modulate the amount of prior knowledge that informs the constructed regions but can, on a different plane, finely modulate their commitment to such information. The result is a versatile certified estimation framework capable of addressing a multitude of problems, ranging from parametric estimation (where the probabilistic guarantees can be issued under no commitment to the prior information) to non-parametric problems (that call for fine exploitation of prior information).

*Key words:* system identification; finite sample system identification; uncertainty quantification; kernel-based non-parametric methods; Gaussian regression

## 1 Introduction

We consider the problem of estimating a function from a finite number of input-output examples $\{x_i, y_i\}$, where each $y_i$ is a noisy measurement of the function evaluated at $x_i$. Popular approaches to this problem include the so called kernel-based methods [27,53,20] that rely on the theory of reproducing kernel Hilbert spaces (RKHS) [5,26]. Notable applications of the theory of RKHS to function approximation and estimation are found in support vector machines [19,58] and regularization networks (RN) [61,48,55]. Given the importance played by RNs in this work, it is worth already recalling their structure. Letting $y = [y_1 \dots y_n]^\top$ and $\mathcal{H}$ be the RKHS of functions $f$ with norm denoted by $\|\cdot\|_{\mathcal{H}}$, RN returns the estimate

$$\hat{f} = \arg\min_{f \in \mathcal{H}} (y - f_{1:n})^\top \Sigma_v^{-1} (y - f_{1:n}) + \|f\|_{\mathcal{H}}^2 \qquad (1)$$

where $f_{1:n} = [f(x_1) \dots f(x_n)]^\top$ and $\Sigma_v$ is a weighting matrix. The estimator balances two terms: the first accounts for the adherence to experimental data, while the second is a penalty term which restores well-posedness. This approach thus enables selecting in a principled manner the entire $f$ from a finite amount of measurements also when $\mathcal{H}$ is infinite-dimensional. Importantly, it also enables recasting regularized system identification as function estimation. In this setting, $f$ represents the input-output relationship induced by a dynamic system where $x_i$ are the past input values. The linear system scenario is then recovered by selecting a linear kernel [46,47].

An important feature of any estimation algorithm is the ability to return uncertainty bounds around the estimates. Given their importance, studies on error bounds

⋆ This paper was not presented at any IFAC meeting. Corresponding author Gianluigi Pillonetto `giapi@dei.unipd.it`

*Email addresses:* `baggio@dei.unipd.it` (Giacomo Baggio), `algo.care@unibs.it` (Algo Carè), `ascampicc@ethz.ch` (Anna Scampicchio), `giapi@dei.unipd.it` (Gianluigi Pillonetto).

and learning rates abound in the machine learning literature. One approach consists of constraining the estimate to a given function space like a RKHS, and bounds are typically computed leveraging stochastic inequalities. Examples of non-asymptotic uncertainty regions built around (1) can be found, e.g., in [54,65,64,62,31]; see also [7], where the study involves a large class of regularization methods. Even if of great theoretical value, such results appear however of limited applicability: the resulting bounds are much conservative and often depend on the unknown function, making them not computable in practice. Recently, error bounds for dynamic systems learning, connected with those obtained in [29], have been also derived [57,8,16]. They can be evaluated before any data is observed but, since they must be valid for all (or most of the) models falling in a particular class, they are often too loose for the particular dynamic system at hand.

Other approaches return bounds that, beyond being non-asymptotic, are also exact, i.e., with the desired inclusion probability. This requires additional assumptions on data generation, like the introduction of prior distributions on $f$. A notable example is the use of a Bayesian framework [23,37] where (1) is interpreted as the solution to a Gaussian regression problem [52]. In fact, the link with Gaussian regression is obtained by modeling the measurement noise and the unknown predictor $f$ as (independent) Gaussian processes [39]. Hence, the posterior density becomes available in closed form and Bayes intervals can be easily computed. Applications here abound: see, e.g., [25,17,50]. However, this approach may be subject to the following limitations (denoted with GR-L):

GR-L1 it is common practice to set $\Sigma_v = \gamma I$ in (1), hence assuming stationary measurement noises. Such model is not adequate, e.g., when data are collected by sensors with different precisions, a situation often encountered in wireless sensor networks [1]. This simplification is often made since the exact calibration of the noise variances is difficult. This problem has been treated in the context of heteroscedastic Gaussian process regression, e.g., assuming that the noise variances smoothly depend on the inputs $\{x_i\}$ [28,41]. Solutions however require the introduction of delicate prior distributions on $\Sigma_v$. Inference then requires using Markov chain Monte Carlo (MCMC) or variational Bayes methods [38,40,33] whose implementation may be non-trivial. For instance, MCMC needs careful tuning of proposal densities and a different design is required for any different adopted prior, e.g., Gaussian [42], Laplacian [44], horseshoe [36] or leading to Bayesian bridge estimation [49].

GR-L2 Gaussian assumptions are often inadequate: many natural phenomena are better represented, e.g., by Laplacian or Student's $t$ distributions. Non-Gaussian distributions can be used to describe outliers that contaminate the measurements or to promote sparsity in the estimation process like in the LASSO [56,67]. Non-Gaussian models can be handled in finite-dimensional settings, but inference typically requires stochastic simulation techniques (like the above mentioned MCMC) that can be computationally demanding and subject to uncertain convergence [51,3,22];

GR-L3 sometimes there is not sufficient information to postulate a parametric form for some distributions, making difficult even to define the posterior.

Another approach to build bounds for linear regression is the sign-perturbed sums (SPS) technique [14]. Following a randomization principle, it constructs guaranteed uncertainty regions for deterministic parametric models in a quasi-distribution free set-up [9,10]. Recently, there have been notable attempts to extend the class of models that SPS can handle. The first line of thought still considers the unknown parameter as deterministic but introduces regularization, see [59,13,45] and also [15], which is a first attempt to move beyond the strictly parametric nature of SPS. A second line of thought enables exploiting some form of prior knowledge at a more fundamental probabilistic level: this is the case of the approaches presented in [45,11], where, however, only knowledge about the symmetry around zero of the parameter densities can be exploited. Overall, a more widespread use of SPS has so far been hindered by the following limitations (denoted with SPS-L):

SPS-L1 computational difficulties and the lack of freedom to include knowledge of some prior distributions. E.g., also in connection with GR-L1, one could know that measurements are collected by different sensors. Some of them could be reliable (providing data corrupted by noises of small variance) while others could be less reliable (possibly returning outliers [32]). This information is important but currently it cannot be included in any SPS algorithm;

SPS-L2 the parametric nature of SPS that prevents computing uncertainty bounds around the estimates returned by RNs equipped with infinite-dimensional RKHSs. Such spaces are adopted in system identification (see Section 7 for an example) and are a key tool in the broader area of statistics and machine learning: in fact, they define important universal models [43] like those induced by Gaussian or spline kernels.

As we shall see, the present paper makes a significant step to overcome all the five aforementioned limitations affecting Gaussian regression and SPS. In particular, we develop a framework that greatly generalizes SPS. It can handle stochastic linear models containing random variables that have either known probability densities (of any form) or just assumed to be symmetric around zero, hence addressing SPS-L1 and connecting with Bayesian statistics. Nevertheless, our approach does

not use the concept of a posteriori density function in that the latter cannot be defined when prior distributions are not completely specified, as allowed by our setting. It returns uncertainty regions that have the desired and exact coverage level in a Bayesian frequentist sense [6], i.e., they have the exact probability to contain the unknown function in many repetitions of the experiment (where each experiment consists of observing a new/independent joint realizations of the noise and the parameters). They are built around the regularized least squares estimates as given in (1) and are easily computable as the union of a finite number of ellipsoids.

From a technical viewpoint, we extend the classic SPS framework [14] and its existing Bayesian generalizations [45,11] by introducing new perturbed versions of the SPS reference function, which are either constructed via sign-perturbations or sampling from the distribution of the parameter and noises (whenever available), depending on the user's choice. These newly introduced test functions allow us to address SPS-L1 and establish a new SPS-based algorithm returning uncertainty regions with exact coverage probability even when prior information on the parameter and noise distributions is exploited. In addition, we propose an extension of our algorithm which, for the first time in the literature, returns exact regions also when regularization and kernel parameters need to be estimated from data and some priors turn out to be wrong.

As illustrated in the paper, our Bayesian frequentists bounds (BFB) find wide applicability and usefulness in many different contexts. For instance, in connection with GR-L2, these include robust and sparse estimation implemented, e.g., by the Bayesian LASSO in which unknown parameters follow a Laplacian distribution [56,44]. In response to SPS-L2 and GR-L3, our technique can handle RNs equipped with infinite-dimensional RKHSs and return exact uncertainty regions just assuming that noise densities are symmetric around zero. This addresses also GR-L1 by providing a novel approach to heteroscedastic Gaussian process regression: exact bounds can be obtained without knowing the dependence of the noise variance on input locations. An important application concerns the construction of uncertainty regions around estimates of dynamic systems.

The paper is organized as follows. Section 2 describes the problem in mathematical terms setting up some useful notation. An estimator which can capture any sampled version of the infinite-dimensional estimate returned by (1) is also introduced. Section 3 provides an informal presentation of the rationale behind the proposed BFB and its evolution from existing SPS approaches. Section 4 describes our algorithm to build BFB and displays the statement of the main Theorem. First applications are illustrated in Section 5. Section 6 describes the extension of our results to the case of regularization and kernel parameters estimated from data and wrong priors. The importance of BFB for linear system identification is then illustrated in Section 7 through a numerical experiment. The Appendix contains the proofs of the main results, some technical details, and additional examples.

## 2 Problem statement

Let us start by formalizing our problem. Data are in the vector $y \in \mathbb{R}^n$ and the measurement model is

$$y = A\theta^0 + B\nu \tag{2}$$

where $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{n\times n}$ are known matrices, both assumed full rank, $\theta^0 \in \mathbb{R}^m$ is the unknown vector while $\nu$ contains the noise components. We model both $\theta^0$ and $\nu$ as (independent) random vectors. The unknown vector $\theta^0$ is the sum of a (known) vector $\mu \in \mathbb{R}^m$ and the output of a linear system described by the $m \times m$ (known, full rank) matrix $C$ fed with noise $\omega$:

$$\theta^0 = \mu + C\omega. \tag{3}$$

Observe that (2) is a standard linear measurement model where $A$ is the regression matrix, $v = B\nu$ is the measurement noise and $\theta^0$ the unknown (random) parameter.

**Remark 1** *The vectors $v$ and $\theta^0$ are defined in terms of the random vectors $\nu$ and $\omega$, respectively, and of the parameters $\mu$, $B$, $C$. The latter parameters add flexibility to the model as they can be used to encode second-order information on $v$ and $\theta^0$. In fact, Equations (2)-(3) offer a general model that captures many relevant situations as special cases. Consider, for instance, the Gaussian regression framework, where*

$$\theta^0 \sim \mathcal{N}(\mu, \Sigma_\theta), \quad v \sim \mathcal{N}(0, \sigma^2 I).$$

*This situation can be described by our model by letting $\omega$ and $\nu$ be (independent and standardized) white Gaussian noises, $\omega, \nu \sim \mathcal{N}(0, I)$, $B = \sigma I$ and $C = \Sigma_\theta^{1/2}$. Similarly, if the covariance of $v$ is $\Sigma_v$, we can simply set $B = \Sigma_v^{1/2}$. The possibility to specify $B$ and $C$ is particularly useful when the distributions of $\theta^0$ and $v$ are only partially known. In fact, with the choice $B = \Sigma_v^{1/2}$ and $C = \Sigma_\theta^{1/2}$, the covariances of $v$ and $\theta^0$ remain equal to $\Sigma_v$ and $\Sigma_\theta$ for all the possible distributions of $\nu$ and $\omega$, provided that they are standardized white.*

In this paper, we will make use of the following mild assumption on the noises $\nu$ and $\omega$.

**Assumption 1** *The components of $\nu$ and $\omega$ are independent and divided into two distinct sets. The first, denoted by $\mathcal{A}$, contains random variables of known probability distribution. The second, named $\mathcal{B}$, contains random variables with probability densities just known to be symmetric around zero.*

Notice from Assumption 1 that our framework departs from a purely Bayesian setting because it does not require complete information about prior distributions: the distributions of some or all the components of $\nu$ and $\omega$ can be just known to be symmetric around zero. We stress that symmetry around zero is satisfied in many common and relevant noise scenarios, such as for independent zero-mean uniform, Laplace, Student's $t$ noises.

Our problem is to build regions, based on the measurements collected in vector $y$, containing the true vector $\theta^0$ with prescribed probability $\alpha \in (0,1)$. Any realization of our regions will be guaranteed to contain the estimate

$$\begin{aligned}\hat{\theta} &= \arg\min_{\theta} (y-A\theta)^\top \Sigma_v^{-1}(y-A\theta)+(\theta-\mu)^\top \Sigma_\theta^{-1}(\theta-\mu) \\ &= \mu + (A^\top \Sigma_v^{-1} A + \Sigma_\theta^{-1})^{-1} A^\top \Sigma_v^{-1}(y-A\mu) \qquad (4)\end{aligned}$$

where matrices $\Sigma_v = BB^\top$ and $\Sigma_\theta = CC^\top$ are invertible by assumption. Although other estimators may be preferable to (4) depending on the intended application and on the available prior information on the noise distributions, it is worth recalling two important facts about (4):

- if the components of $\nu$ and $\omega$ are all mutually uncorrelated, with zero mean and unit variance, the covariances of the stochastic terms $B\nu$ and $C\omega$ in (2) and (3) correspond exactly to $\Sigma_v$ and $\Sigma_\theta$, respectively, and $\hat{\theta}$ thus becomes the minimum variance linear estimator of $\theta^0$, see, e.g., [2];
- (4) can capture any sampled version of the infinite-dimensional estimate returned by (1). In fact, consider the estimator $\hat{f}$ given by (1), function of the measurements in $y$ collected on the inputs $\{x_i\}_{i=1}^n$. Consider other $n^* - n$ generic input locations $\{x_i\}_{i=n+1}^{n^*}$ on which we want to perform prediction and let $\mathbf{K}$ be the kernel matrix of dimension $n+n^*$ with $(i,j)$-entry $K(x_i,x_j)$. Then, defining

$$A = [I_n \; 0_{n\times n^*}], \quad \mu = 0_{(n+n^*)\times 1} \quad \text{and} \quad \Sigma_\theta = \mathbf{K}, \qquad (5)$$

where $I_n$ is the $n$-dimensional identity matrix and $0_{p\times q}$ the $p\times q$ zero matrix, and using the Representer Theorem [53, Theorem 4.2] one obtains that

$$\hat{\theta} = \hat{f}_{1:n+n^*} := [\hat{f}(x_1) \; \hat{f}(x_2) \; \ldots \; \hat{f}(x_{n+n^*})]^\top. \qquad (6)$$

Ultimately, we point out that in many situations it is important to explicitly include a scale factor $\gamma > 0$ in the penalty term of (1). In our setting, this is simply achieved by rewriting $C$ as $C = \gamma^{-1/2}\tilde{C}$.

## 3 Overview of SPS and the new BFB

In this section, we review the main idea of SPS and outline the evolution that has led to the present work.

The original set-up of SPS was first presented in [14] and is the following. Let us consider the measurements model (2), where $B = I_n$ (this is without loss of generality because, if $B \neq I_n$, all the terms can be multiplied by $B^{-1}$ on the left); it is here further assumed that
(i) all the components in the noise vector $\nu$ are independent, with probability distributions that are symmetric around 0;
(ii) vector $\theta^0$ is deterministic, fixed yet unknown.
The SPS approach is set in a frequentist framework, and its goal is to return a confidence region, denoted by $\Theta_{SPS}(\mathrm{y})$, containing the true value $\theta^0$ with exact user-chosen probability.

We recall the following important facts that have been proven for this set-up in [14] and [63].

*Fact 1 ([14], Theorem 1)*: The confidence region $\Theta_{SPS}(y)$ returned by the SPS algorithm contains $\theta^0$ with probability $1-q/r$, where $q$ are $r$ are user-chosen positive integers such that $q < r$. That is, $\mathbb{P}(\theta^0 \in \Theta_{SPS}(y)) = 1-q/r$.
*Fact 2 ([14], Theorem 2)*: $\Theta_{SPS}(y)$ is built around the least-squares estimate $\hat{\theta}_{LS}$, i.e., the minimizer of the quadratic function $(y-A\theta)^\top(y-A\theta)$.
*Fact 3 ([14], Appendix B)*: One can express $\Theta_{SPS}(y)$ as a union of intersections of ellipsoids.
*Fact 4 ([63]):* Under nonvanishing excitation, and suitable restrictions on the growth rate of the moments of the noise and the regressors, the SPS algorithm is strongly consistent (see Theorem 2 in [63] for details). Moreover, when the noise is i.i.d. with bounded 4th moment, the shape of the region is asymptotically optimal (see Theorem 3 in [63] and the discussion therein).

The SPS region $\Theta_{SPS}(y)$ is formally defined as the set of candidate parameters $\theta$ that pass an *inclusion test*, which we now describe. The test is based on the so-called normal equation $A^\top(y-A\theta) = 0$, whose solution is $\hat{\theta}_{LS}$. Denoting by $A(t,:)$ the $t$-th row of matrix $A$, the normal equation can also be written as

$$\sum_{t=1}^{n} A(t,:)^\top (y_t - A(t,:)\theta) = 0. \qquad (7)$$

Denote by $H_0(\theta)$ the left-hand side of (7), and consider $r-1$ "sign-perturbed" versions of $H_0(\theta)$ defined as

$$H_i(\theta) = \sum_{t=1}^{n} A(t,:)^\top \left(\varsigma_{t,i}(y_t - A(t,:)\theta)\right), \qquad (8)$$

where $\varsigma_{1,i}, \ldots, \varsigma_{n,i}$, $i = 1, \ldots, r-1$, are i.i.d. Rademacher random variables (i.e., each $\varsigma_{t,i}$ takes value $+1$ or $-1$ with equal probability; we informally denote such variables with $\pm$). The inclusion test prescribes to compute the Euclidean norm $\|H_i(\theta)\|$ for all $i = 0, \ldots, r-1$ and to sort these values: a candidate $\theta$ is excluded from $\Theta_{SPS}(y)$ if and only if $\|H_0(\theta)\|$ is among the $q$ highest values.

Let us now briefly connect this construction with Facts 1÷4. Considering that $\|H_i(\hat{\theta}_{LS})\|$ is positive for all $i \neq 0$, while $\|H_0(\hat{\theta}_{LS})\| = 0$ holds by construction, one has that $\hat{\theta}_{LS} \in \Theta_{SPS}(y)$, thus showing Fact 2. Fact 1 follows by proving that, when $\theta = \theta^0$, $\|H_0(\theta^0)\|, \ldots, \|H_{r-1}(\theta^0)\|$ are identically distributed, and that the probability that $\|H_0(\theta^0)\|$ is the $j$-th in the ranking is $1/r$ for each $j = 1, \ldots, r$. In fact, this implies that the ranking of $\|H_0(\theta^0)\|$ is smaller than $q$ with probability $1 - q/r$, which is also the probability that $\theta^0 \in \Theta_{SPS}(y)$. Regarding, instead, the shape of the region, it is important to note that wrong values of $\theta$ tend to be excluded from the SPS region because, when $\theta \neq \theta^0$, the random signs induce cancellations in the perturbed sums (8) and $\|H_0(\theta)\|$ tends to be larger than any other $\|H_i(\theta)\|$. In this line, the consistency of the algorithm can be proven. At this point, we must observe that the standard SPS algorithm relies on an inclusion test that slightly differs from the one that we have just described. Precisely, in standard SPS, $H_0(\theta), \ldots, H_{r-1}(\theta)$ are pre-multiplied by the "shaping matrix" $(A^\top A)^{-\frac{1}{2}}$ before taking norms and ranking them (in fact, $(A^\top A)^{-\frac{1}{2}}$ equalizes the components of $H_0(\theta)$ in the sense that $(A^\top A)^{-\frac{1}{2}} H_0(\theta^0)$ has unit covariance when $\nu$ has unit covariance). The "shaping matrix" improves the shape of the SPS region and plays a role in the proofs of Facts 3 and 4, see Section IV.A in [14] and the proof of Theorem 3 in [63] for more details.

Following the analysis of [13,45,60], SPS can be easily modified so that the regularized estimate $\hat{\theta}$ in (4) takes the place of the least squares estimate $\hat{\theta}_{LS}$. To this purpose, we note that $\hat{\theta}$ is the solution to the equation $A^\top(y - A\theta) + \Sigma_\theta^{-1}(\mu - \theta) = 0$. Proceeding, *mutatis mutandis*, as in the least squares case, and recalling that $\Sigma_\theta^{-1} = \sum_{k=1}^m C^{-1}(k,:)^\top C^{-1}(k,:)$, we can redefine the test functions as

$$\tilde{H}_0(\theta) = A^\top(y - A\theta) + \Sigma_\theta^{-1}(\mu - \theta) \tag{9}$$

$$\begin{aligned}\tilde{H}_i(\theta) = &\sum_{t=1}^n A(t,:)^\top \left(\varsigma_{t,i}(y_t - A(t,:)\theta)\right) \\ &+ \sum_{k=1}^m C^{-1}(k,:)^\top (C^{-1}(k,:)\mu - C^{-1}(k,:)\theta),\end{aligned} \tag{10}$$

(the shaping matrix can also be redefined as $(A^\top A + (C^{-1})^\top C^{-1})^{-\frac{1}{2}}$), all the rest being unchanged. Importantly, in the so-obtained regularized SPS, $\mu$ and $\Sigma_\theta = CC^\top$ are just user-chosen parameters that help improve the stability of the estimate; as such, they are not subject to random sign perturbations. Although they can be used to bias the estimation algorithm towards values of $\theta$ that are deemed to be more likely, a bad specification of them can only affect the size of the region, while the coverage probability remains always equal to $1 - q/r$. This robustness against the user's wrong beliefs is a notable property but comes at a price: it limits the extent to which prior knowledge about $\theta^0$ can be exploited by the algorithm. An illustration of this limitation is offered by Example 1 in Section 9.1 of the Appendix.

The new approach of this paper, BFB, extends the original SPS idea in two ways. Firstly, following the preliminary studies in [45,11], $\theta^0$ is no more deterministic but is random (as is typical of a Bayesian approach): namely, model (2) is complemented by model (3) that accounts for the variability of $\theta^0$ through the random $\omega$, and the probability of including $\theta^0$ is now computed with respect to the joint distribution of $\nu$ and $\omega$. Secondly, we modify the test functions to accommodate the case where knowledge on the distribution of $\nu$ and $\omega$ is available and the user is willing to commit to it, either partially or entirely. At an algorithmic level, the reference function $\tilde{H}_0(\theta)$ for BFB is defined as in (9), while the perturbed versions are instead

$$\tilde{H}_i(\theta) = \sum_{t=1}^n A(t,:)^\top \tilde{\nu}_{t,i}(\theta) + \sum_{k=1}^m C^{-1}(k,:)^\top \tilde{\omega}_{k,i}(\theta), \tag{11}$$

where $\tilde{\nu}_{t,i}(\theta)$ and $\tilde{\omega}_{k,i}(\theta)$ are user-generated random elements. If the user wants to commit only to the belief that $\nu_t$ or $\omega_k$ are symmetrically distributed (i.e., they belongs to $\mathcal{B}$ according to Assumption 1), then the random elements $\tilde{\nu}_{t,i}(\theta)$ or $\tilde{\omega}_{k,i}(\theta)$ are constructed by means of sign-perturbations as $\tilde{\nu}_{t,i}(\theta) = \pm(y_t - A(t,:)\theta)$ or $\tilde{\omega}_{k,i}(\theta) = \pm(C^{-1}(k,:)\mu - C^{-1}(k,:)\theta)$. On the other hand, if, for some (or even all) $t$ and $k$, the distributions of $\nu_t$ or $\omega_k$ are known (i.e., $\nu_t$ or $\omega_k$ belong to $\mathcal{A}$ according to Assumption 1) and the user wants to exploit this knowledge, then $\tilde{\nu}_{t,i}$ or $\tilde{\omega}_{k,i}$ are generated as fresh, independent samples according to the known distributions. This latter extension of SPS which accounts for distributions that are fully known for some (or even all) $\nu_t$ and $\omega_k$, is introduced and studied for the first time in the present work.

The last ingredient of the BFB algorithm as presented in the next section is the choice of the parameter $q$, which is no more a free parameter as in SPS but is always set equal to 1. Thanks to this choice, the computation of BFB does not involve intersections of ellipsoids as in SPS (see Fact 3 above), but only unions: see (15) in the BFB Algorithm 1. All the algorithmic details and the relevant finite-sample properties of BFB are presented in the next section.

## 4 Bayesian frequentist bounds

We are now ready to describe the proposed algorithm. It will return a region containing the unknown vector $\theta^0$ with exact probability $\alpha = 1 - 1/r$, where $r$ is a positive integer. Here, with an eye to practical applications, we focus on the computation of 95% ($r = 20$) and 99% ($r =$

100) uncertainty bounds, however, the generalization of our results to any choice of $r$ is trivial.

First, we introduce some preliminary notation. We define $\Omega \in \mathbb{R}^{N\times m}$, $z \in \mathbb{R}^N$, and $\epsilon \in \mathbb{R}^N$, where $N = n + m$, respectively, as

$$\Omega = \begin{bmatrix} B^{-1}A \\ C^{-1} \end{bmatrix}, \quad z = \begin{bmatrix} B^{-1}y \\ C^{-1}\mu \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \nu \\ \omega \end{bmatrix}. \tag{12}$$

For $i = 1, \dots, r-1$ and $t = 1, \dots, N$, we use $\{\xi_t^i\}$ to denote a sequence of $(r-1)N$ independent random variables satisfying

$$\begin{cases} \mathbb{P}(\xi_t^i = 0) = 1 & \text{if } \epsilon_t \in \mathcal{A}, \\ \mathbb{P}(\xi_t^i = -1) = \mathbb{P}(\xi_t^i = 1) = \frac{1}{2} & \text{if } \epsilon_t \in \mathcal{B}. \end{cases} \tag{13}$$

Further, for $i = 1, \dots, r-1$, we use $\zeta^i$ to indicate independent random vectors whose components $\zeta_t^i$, $t = 1, \dots, N$, are samples from the distribution of $\epsilon_t$ if $\epsilon_t \in \mathcal{A}$, and zero with probability one if $\epsilon_t \in \mathcal{B}$. The quantities introduced above allow us to define randomized copies of the vector $\epsilon$: its components in $\mathcal{B}$ will be perturbed by random signs, while those in $\mathcal{A}$ will be drawn from their prior distribution. Finally, we let $\pi = (\pi(0), \pi(1), \dots, \pi(r-1))$ denote a uniformly distributed random permutation of the set $\{0, 1, \dots, r-1\}$ whose (merely technical) role is discussed in Lemma 1 of Section 9.2 of the Appendix.

**Theorem 1 (Exact inclusion probability)** *Consider the model in* (2)-(3) *and Assumption 1. Then, the Bayesian frequentist region* $\Theta(y)$ *defined through Algorithm 1 is a union of convex sets, contains* (4) *if non-empty, and satisfies*

$$\mathbb{P}(\theta^0 \in \Theta(y)) = \begin{cases} 0.95, & \text{if } r = 20, \\ 0.99, & \text{if } r = 100. \end{cases} \tag{16}$$

The proof of Theorem 1 is deferred to Section 9.2 of the Appendix.

The probability in (16) is unconditional, so $\Theta(y)$ is a Bayesian frequentist error bound, which can be interpreted as follows: consider $M$ independent experiments where, in each of them, we observe a joint realization of $\theta^0$ and of $y$ generated through (2)-(3). This results in a sequence of $M$ distinct regions $\Theta(y)$ of coverage level $\alpha$. Then, as $M$ goes to infinity, the frequency with which each regions contains the corresponding value of $\theta^0$ will tend to $\alpha$.

In practice, to compute the region $\Theta(y)$ in (15) one has to run Algorithm 1 using a realization of the random elements $z$, $\{\xi_t^i\}$, $\{\zeta_t^i\}$, and $\pi$. The main cost of Algorithm

**Algorithm 1** Bayesian frequentist uncertainty region of probability $\alpha = 1 - 1/r$

Compute

$$Q_i = \frac{1}{N}\sum_{t=1}^{N} \xi_t^i \Omega(t,:)^\top \Omega(t,:),$$
$$\psi_i = \frac{1}{N}\sum_{t=1}^{N} \xi_t^i \Omega(t,:)^\top z_t + \frac{1}{N}\sum_{t=1}^{N} \Omega(t,:)^\top \zeta_t^i$$

for $i = 1, \dots, r-1$, where $\Omega(t,:)$ is the $t$-th row of $\Omega$ (thus $\Omega(t,:)^\top$ is a column vector).

Define the sets

$$\mathcal{E}_i = \{\theta \in \mathbb{R}^m \ : \ \theta^\top A_i \theta + 2\theta^\top b_i + c_i \leq_{\pi,i} 0\}, \tag{14}$$

for $i = 1, \dots, r-1$, where

$$R_N = \frac{1}{N}\Omega^\top\Omega, \ A_i = R_N - Q_i R_N^{-1} Q_i,$$
$$b_i = Q_i R_N^{-1}\psi_i - R_N\hat{\theta}, \ c_i = \hat{\theta}^\top R_N \hat{\theta} - \psi_i^\top R_N^{-1}\psi_i,$$

with $\hat{\theta}$ defined as in (4), and "$\leq_{\pi,i}$" stands for "$\leq$" if $\pi(0) < \pi(i)$, and for "$<$" otherwise.

Return the region

$$\Theta(y) = \bigcup_{i=1}^{r-1} \mathcal{E}_i. \tag{15}$$

1 comes from the computation of the $r-1$ convex regions $\mathcal{E}_i$ that define $\Theta(y)$. The main cost of computing such regions in turn arises from the computation of $R_N$, which requires the multiplication of two matrices of dimensions $m \times N$ and $N \times m$, where $N = n + m$. This yields an overall complexity of $O(m^2 N)$, which scales cubically in the dimension $m$ of the unknown vector and linearly in the number of measurements $n$.

Importantly, the probability of $A_i$ to be singular is normally negligible in real applications (see Section 9.4 of the Appendix for details), so that (the closure of) the region (15) is the union of $r-1$ ellipsoids of the form

$$\mathcal{E}_i = \{\theta \in \mathbb{R}^m \ : \ (\theta - \bar{b}_i)^\top A_i (\theta - \bar{b}_i) \leq \bar{c}_i\}$$

for $i = 1, \dots, r-1$, where $\bar{b}_i = -A_i^{-1} b_i$ and $\bar{c}_i = -c_i + b_i^\top A_i^{-1} b_i$, and always contains the estimate (4). From the latter expression, useful information about the region (15) can be easily extracted. In particular, it follows that lower and upper bounds of the components of the vectors

in $\Theta(y)$ can be computed as follows:

$$\begin{aligned}\theta_{\min} &= \min_{i=1,\dots,r-1} \left\{ \bar{b}_i - (\bar{c}_i \operatorname{diag}(A_i^{-1}))^{\frac{1}{2}} \right\}, \\ \theta_{\max} &= \max_{i=1,\dots,r-1} \left\{ \bar{b}_i + (\bar{c}_i \operatorname{diag}(A_i^{-1}))^{\frac{1}{2}} \right\},\end{aligned} \tag{17}$$

where $\operatorname{diag}(A_i)$ denotes the vector of diagonal entries of $A_i$ and the $(\cdot)^{\frac{1}{2}}$, min, max operators are applied component-wise. Finally, the region can be efficiently reconstructed in sampled form. In fact, one can draw independent and uniform realizations from each ellipsoid, e.g., via the algorithm described in [18]. The sampled-form representation of $\Theta(y)$ and the component-wise bounds in (12)-(13) serve as useful tools to graphically represent $\Theta(y)$, especially when $m$ is large, and they will be used in the numerical experiments described in the next section.

## 5 Numerical experiments

To demonstrate the relevance and applicability of our Bayesian frequentist framework, we present two numerical applications of our uncertainty bounds.

### *5.1 BFB with completely specified probability distributions*

We consider three examples where the probability distribution on $\theta^0$ and the noise distributions are completely specified, and we commit to this specification when we compute the probability of the region. Thus, referring to Assumption 1, all the noises $\nu$ and $\omega$ belong to the set $\mathcal{A}$. In this case, uncertainty bounds could be extracted from the posterior $\mathbf{p}(\theta^0|y)$. Typically, for non Gaussian densities, one has to resort to stochastic simulation schemes (like MCMC) whose design depends on the involved distributions and can be far from trivial. We now show that, when data come from the model specified by (2) and (3), Algorithm 1 returns alternative and informative uncertainty regions in a very efficient way: in all tests the computational time never exceeds one tenth of a second. Before describing the examples, we mention that when $\nu$ and $\omega$ contain i.i.d. Gaussian variables (full Gaussian case), the uncertainty regions proposed in this paper and Bayes credible regions are often comparable in size. In particular, it can be shown that the two regions coincide as $m$, the dimension of $\theta^0$, tends to infinity.

*Laplacian noise.* Let $\theta^0$ and $\nu$ be, respectively, Gaussian and Laplacian random vectors. Their components are independent, with zero-mean and unit variance:

$$\begin{gathered}\theta_i^0 \sim \mathcal{N}(0,1), \quad \nu_j \sim \operatorname{Lap}(0,1), \\ i = 1,\dots,m,\ j = 1,\dots,n.\end{gathered}$$

This is an important model: Laplacian distributions are often used to describe stochastic sources that can contaminate the data with outliers, e.g., see [35,34,4]. Their use can define more robust bounds accounting for unexpected noise model deviations. Consider an example where the data set size is $n = 200$, the dimension of $\theta^0$ is $m = 20$, and the entries of the regression matrix $A$ were set equal to values randomly drawn according to independent zero-mean Gaussian distributions of unit variance. One realization of $y$ is in the top left panel of Figure 1 while the bottom left panel displays that of $\theta^0$ (solid line). Algorithm 1 was used to build the uncertainty region setting in (3) all the entries of $\mu$ to zero and $B = I_n, C = I_m$. The bottom left panel of Figure 1 displays the 95% bounds reporting the upper and lower limits of (17) using dashed lines. They are very informative and give a clear picture about the information provided by the outputs. Differently from MCMC, they are obtained without the need of defining any proposal density. In Section 9.5 of the Appendix, a detailed comparison between BFB and Bayesian intervals computed via MCMC is provided.

*Laplacian and Gaussian noise.* To illustrate the versatility of our approach, consider the same experiment except that the first 50 outputs are generated in a different way. They depend only on the first 5 components of $\theta^0$. Specifically, the regression matrix $A$ is as before except for its first 50 rows which are now set to zero from column 6 to column 20. The measurements noise is now white and Gaussian with small variance 0.01. This means that measurements contain more information to reconstruct the first portion of the parameter vector $\theta^0$ than to estimate the last 15 components. Data and bounds by Algorithm 1 with $B = I_n, C = I_m, \mu = 0_{m\times 1}$ are in the middle panels of Figure 1. One can see that BFB well accounts for the new data by returning tighter bounds around the first 5 components of $\hat{\theta}$.

*Bayesian LASSO.* LASSO is a popular technique for linear regression that adopts the $\ell_1$ penalty to regularize coefficients and induce sparse solutions [56]. Building uncertainty bounds around LASSO estimates is however difficult and (approximate) error estimators like, e.g., those described in [56,21] often do not return satisfactory results. An alternative developed in [44] is to use MCMC exploiting a stochastic intepretation of the problem, the so called Bayesian LASSO, where parameters follow a Laplacian distribution. For our example, let now $\theta^0$ and $\nu$ contain, respectively, independent Laplace and Gaussian variables with zero-mean and unit variance, i.e. $\theta_i^0 \sim \operatorname{Lap}(0,1)$ and $\nu_j \sim \mathcal{N}(0,1)$. As in the previous case, the data set size is $n = 200$, the dimension of $\theta^0$ is $m = 20$ and the regression matrix $A$ is the same as in the Laplacian noise case. Data and results by Algorithm 1 with $B = I_n, C = I_m, \mu = 0_{m\times 1}$ are in the right panels of Figure 1. BFB bounds appear somewhat informative also for variable selection purposes where one wants to

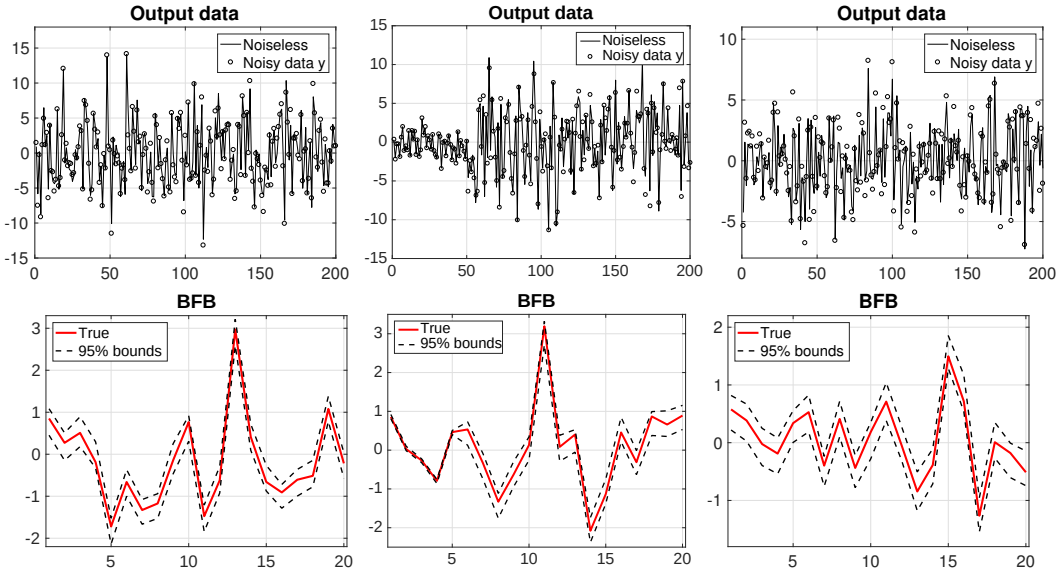


Fig. 1. Example of output data $y$ and BFB for $\theta^0$ for *Laplacian noise* (left panels), *Gaussian and Laplacian noise* (middle) and *Bayesian LASSO* (right).

assess the actual influence of the components of $\theta^0$ on $y$.

### 5.2 *BFB for heteroscedastic Gaussian regression with uncertain noise distributions*

We consider two heteroscedastic Gaussian regression problems. The function $f$ is a continuous-time normal process to be estimated from $\{x_i, y_i\}_{i=1}^n$. The measurement noise is Gaussian with variances that depend on the input $\{x_i\}$. Two kinds of bounds built around (1) will be compared: those classical, that assume both $f$ and the noise $\nu$ to be Gaussian; and the new BFB, where $f$ is Gaussian but we assume only that noise densities are symmetric around zero. Thus, according to Assumption 1, we have $\omega \in \mathcal{A}$ and $\nu \in \mathcal{B}$; bounds are then computed by Algorithm 1 on the 500 inputs $\{0.2, 0.4, \dots, 100\}$ using (5). BFB will be displayed in sampled form by drawing 10000 realizations uniformly distributed over the ellipsoids forming the uncertainty region.

*Cubic spline regression.* Let $f$ be two-fold integration of white Gaussian noise of unit variance:

$$f \sim \mathcal{N}(0, K),$$
$$K(x_i, x_j) = \frac{x_i x_j \min(x_i, x_j)}{2} - \frac{\min(x_i, x_j)^3}{6}.$$

This fundamental model introduces in (1) the regularizer $\|f\|_{\mathcal{H}}^2 = \int \ddot{f}^{\,2}(x)\,\mathrm{d}x$ and leads to cubic smoothing splines [61]. The $y_i$ are collected over the 80 inputs $\{1, 2, \dots, 20\} \bigcup \{41, 42, \dots, 100\}$. Measurements are affected by white Gaussian noises of variances $\sigma_i^2$. We simulate a situation where wrong information on such variances is available due, for example, to some sensors being broken. Precisely, all the sensors collecting the $y_i$ are assumed to have the same nominal precision, i.e. $\sigma_i^2 = \sigma^2 = 2.5$ for all $i$; instead, data $\{y_i\}_{i=1}^{20}$ are more noisy, namely $\sigma_i^2 = 50\sigma^2$, $i = 1, \dots, 20$, as if the first 20 sensors were broken. Figure 2 plots a realization of $f$ (red line), the data $y$ and the estimate (1) obtained with $\gamma^{-1} = \sigma^2$ (black line). At the beginning, the estimate suffers of the large measurement errors and predictions are poor over $[20, 40]$ where no data are available. The left panel of the same figure displays 95% Gaussian bounds. They do not contain the true function: assuming constant noise variance, the uncertainty is underestimated. They could be refined by estimating the $\sigma_i^2$ from data but such calibration is difficult. Our new framework provides an important alternative: the same Gaussian prior on $f$ is adopted but the noise components of $\nu$ are now just assumed independent with densities symmetric around zero. The right panel of Figure 2 plots the new bounds. They are able to contain the true function and give important insights on the actual information contained in the training data. Differently from the Gaussian bounds, they clearly reveal that the estimate is more uncertain where data were less informative.

*Gaussian kernel regression.* The function $f$ is now the realization of a normal process with covariance equal to the Gaussian kernel $K(x_i, x_j) = \exp\{-\left(0.01(x_i - x_j)^2\right)\}$. The $y_i$ are collected over the inputs $\{1, 2, \dots, 100\}$. The Gaussian noises have variances described by the stochastic volatility model used in financial time series and described in [40, Section 6.3.1]. Variances realizations are in the left panel of Figure 3, while the middle and right panels show the realization of $f$ (red line). The Gaussian bounds (middle panel) are obtained having full knowledge of noise statistics, i.e., $\Sigma_v$ is set to the diagonal matrix with entries given by the variances

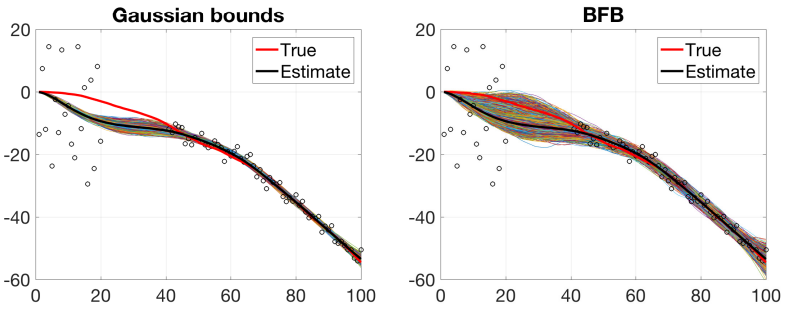


Fig. 2. *Cubic spline kernel regression.* Unknown function $f$ (red line), noisy data ($\circ$), 95% Gaussian bounds with (wrong) constant variance $\sigma^2 = 2.5$ (left) and 95% BFB that uses only knowledge regarding the symmetry of noise probability densities (right).

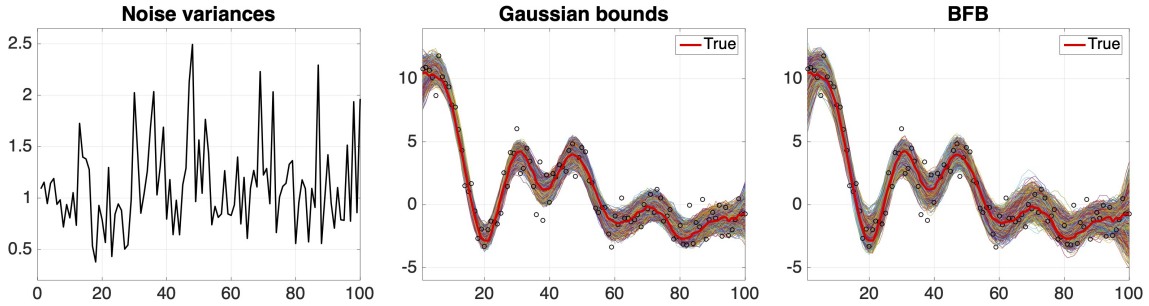


Fig. 3. *Gaussian kernel regression.* Noise variances as function of input locations (left), unknown function $f$ (red line), noisy data ($\circ$), 95% Gaussian bounds with perfect variances knowledge (middle) and 95% BFB with knowledge only on symmetric noises distributions (right).

reported in the left panel. BFB (right panel) instead only exploits information on symmetry of noises distributions and computes the bounds by setting $B$ in (2) to the identity matrix. Remarkably, BFB are similar to the Gaussian bounds.

## 6 BFB handling uncertainty in the priors

In many applications the matrix $C$ that defines the prior on $\theta^0$ in (3) depends on a hyperparameter vector $\eta$ that needs to be estimated from data. This includes the important situation where the regularization parameter $\gamma$ and possibly also some kernel parameters in (1) are unknown and need to be determined from data. Another fundamental issue is that the prior on the function to estimate is never perfect in practice and one would like to achieve bounds robust also with respect to model misspecification. Then, our aim is now to construct uncertainty regions around the estimate (4) with the exact coverage level even if the prior on $\theta^0$ is wrong and irrespective of the way $\eta$ is estimated. As already pointed out, only if the prior matches the true function properties (e.g., smoothness) the regularizer will be useful to reduce the uncertainty associated to the estimates. We will make use of the following assumption.

**Assumption 2** *Assumption 1 still holds but matrix $B$ in (2) is diagonal and matrix $C$ in (3) can depend on an unknown hyperparameter vector $\eta$ that needs to be estimated from data.*

Now, let us assume that the first $\bar{n} < n$ components of the data vector $y$, namely $\bar{y} = [y_1, \dots, y_{\bar{n}}]^\top$, are used to estimate $\eta$ by any calibration procedure, e.g. cross validation or empirical Bayes [32]. We also need to redefine some objects previously introduced in Section 4.

The $(r-1)N$ independent random variables $\{\xi_t^i\}$ are now redefined as

$$\begin{cases} \mathbb{P}(\xi_t^i = 0) = 1 & \text{if } t \in (\bar{n}, n] \text{ and } \epsilon_t \in \mathcal{A}, \\ \mathbb{P}(\xi_t^i = -1) = \mathbb{P}(\xi_t^i = 1) = \frac{1}{2} & \text{if } t \in (\bar{n}, n] \text{ and } \epsilon_t \in \mathcal{B}, \\ \mathbb{P}(\xi_t^i = 1) = 1 & \text{if } t \in [1, \bar{n}] \cup (n, N], \end{cases} \tag{18}$$

for $i = 1, \dots, r-1$ and $t = 1, \dots, N$. Moreover, the $\zeta^i$ now indicate independent random vectors whose components $\zeta_t^i$, $t = 1, \dots, N$, are independent samples from the distribution of $\epsilon_t$ if $\epsilon_t \in \mathcal{A}$ and $t \in (\bar{n}, n]$, and zero with probability one otherwise. This construction implies that only the last $n - \bar{n}$ components of the output vector $y$ are considered as stochastic and are affected by random quantities in the algorithm (they are either multiplied by random signs or replaced by fresh realization of $\nu_t$ from its known distribution), while $\bar{y}$ and $\theta^0$ are treated as deterministic quantities.

We call Algorithm 2 the version of Algorithm 1 with $\{\xi_t^i\}$

and $\{\zeta_t^i\}$ redefined as above. The following result then holds (see Section 9.3 of the Appendix for the proof).

**Theorem 2** *Consider the model in* (2)-(3) *and Assumption 2. Then, the region defined through Algorithm 2 is a union of ellipsoids, contains* (4) *if non-empty, and satisfies*

$$\mathbb{P}(\theta^0 \in \Theta(y) \mid \bar{y}, \theta^0) = \begin{cases} 0.95, & \text{if } r = 20, \\ 0.99, & \text{if } r = 100. \end{cases} \quad (19)$$

The fundamental difference between (16) and (19) is that now the region's probability is exact even if conditioned on a realization of $\bar{y}$ and $\theta^0$. Hence, Algorithm 2 returns an exact uncertainty region even when $\eta$ is estimated from data and/or the prior on $\theta^0$ is completely wrong.

**Remark 2** *Assume that the distributions of the independent components of $\nu$ are just known to be symmetric around zero, and matrix $B$ depends on a scale factor estimated using $\bar{y}$. Then, following the same arguments carried out above, Theorem 2 still holds. This fact is relevant since calibration of the covariance of $B\nu$ can be important to make* (4) *close to the linear minimum variance estimator and to suitably shape the uncertainty region.*

## 7 System identification experiment

We now propose a numerical test for the scenario proposed in Section 6.

Let us assume now that $\theta^0$ is the impulse response of a linear dynamic system to be estimated from input-output data. Differently from all the previous case studies, $\theta^0$ is not drawn from any probability distribution: it is a deterministic 100-dimensional vector whose values are represented in Figure 4 (red line). In what follows, the dimension $m$ of $\theta^0$ is set to 100, leaving no room for undermodeled dynamics. [1]

The system input is white Gaussian noise of variance $10^3$ filtered by the transfer function $1/(z-0.5)$. Its realizations define the (Toeplitz) matrix $A$ in (2). The noise $\nu$ is white and Gaussian with variance $\sigma^2 = 1$. Data set size is $n = 1000$.
To estimate $\theta^0$, one can use (4) with the so-called stable spline kernel that includes smooth-exponential decay information [46,47]. With such a kernel, the regularization matrix is $\bar{K}$ with entries $[\bar{K}]_{ij} = \beta^{\max(i,j)}$, where $0 \leq \beta < 1$ regulates how fast the impulse response goes to zero. The estimates $\hat{\beta}$ and $\hat{\gamma}$ of the decay rate $\beta$ and of the regularization parameter $\gamma$ (that here corresponds to the inverse of the kernel scale factor) are obtained by generalized cross validation (GCV) [30,61] using the first 200 data. The impulse response estimation is then obtained by (4) setting $B = I_n$, $\mu = 0$ and $C = \hat{\gamma}^{-1/2}\bar{K}^{1/2}$ with $\bar{K}$ built using $\hat{\beta}$. The estimate is displayed in Figure 4 (black line) and appears very close to truth (red and black lines are very similar). We will build two different kinds of bounds around the stable spline estimate with the desire to have a 95% coverage level. Specifically, we will compare the performance of Algorithms 1 and 2 when the prior on $\theta^0$ is wrong. As regards measurement noises, we will build both regions just assuming that all components of $\nu$ belong to the set $\mathcal{B}$ defined in Assumption 1. Not having access to prior knowledge on the noise distribution prevents from computing Bayesian posterior bounds, which are the state-of-the-art for kernel-based system identification [25,17,50].

First, we adopt Algorithm 1 modelling $\theta^0$ as a zero-mean Gaussian random vector with covariance $\hat{\gamma}^{-1}\bar{K}$, i.e., Algorithm 1 runs with $A, B, C, \mu$ defined above and assuming $\omega$ white Gaussian noise of unit variance. BFBs by Algorithm 1 are reported in sampled form in the left panel of Figure 4 using 5000 realizations. The bounds are very tight and informative: all the realizations are close to the red line. However, there is no guarantee that the uncertainty region (15) contains the truth with the desired confidence level. In fact, let us think of many repeated experiments where $\theta^0$ is the same deterministic vector as shown in Figure 4 (red line) and only new noise realizations are drawn: since the prior on the impulse response is wrong (the true distribution of $\theta^0$ is concentrated, which does not match the way in which we have defined $\mu$, $C$ and $\omega$), the confidence level of the region returned by Algorithm 1 could be lower than the desired 95%. This is shown by a Monte Carlo test of 1000 runs: the regions built with Algorithm 1 contained $\theta^0$ only in 64% of the trials, thus highlighting the weakness of this approach.

We can now resort to Algorithm 2, setting $\bar{y}$ to the vector containing the first 200 outputs (those used to estimate $\beta$ and $\gamma$). For the first time in the literature, to the best of our knowledge, we obtain an uncertainty region with exact 95% coverage level even if the hyperparameters have been determined from data and the prior on $\theta^0$ is not correct. BFB are displayed in sampled form in the right panel of Figure 4. They are still built around the stable spline estimate, do not largely differ from the previous ones but partially rebel against the prior by becoming more conservative (this is especially evident looking at the impulse response's tail). In this way, they are now able to contain $\theta^0$ with the desired coverage level. A Monte Carlo test of 1000 runs reveals that the 95% region returned by Algorithm 2 includes the true impulse response 95.1% of the times, confirming the theory.

[1] BFB can be used to identify more general systems than finite impulse responses, provided that they can be expressed in linear regression form: see, e.g., the examples in Section 2.B in [14].

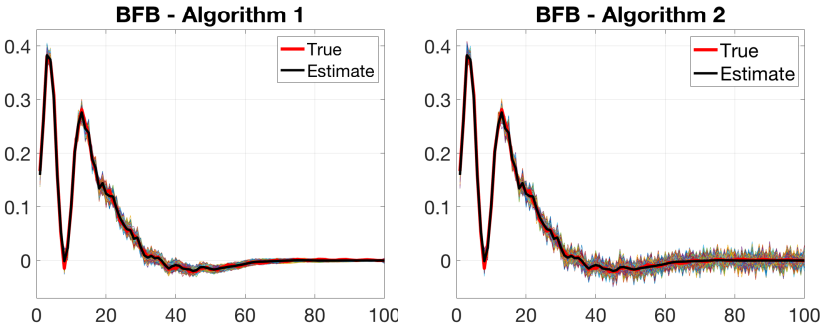


Fig. 4. True deterministic impulse response $\theta^0$ (thick red line) and stable spline estimate (black). Left panel: BFB via Algorithm 1. Measurement noises are assumed independent with symmetric distributions around zero, i.e., $\nu \in \mathcal{B}$. The distribution of the unknown $\theta^0$ is (erroneously) modeled as Gaussian with hyperparameters estimated from data. Bounds are quite informative, but they are not guaranteed to contain the true impulse response with the desired coverage level due to the wrong information on the prior. Right panel: BFB via Algorithm 2 under the same assumptions on noise generation. The bounds are more conservative, but now they contain $\theta^0$ with the desired probability of inclusion (in the frequentist sense) even if hyperparameters are estimated from data the prior of $\theta^0$ is misspecified.

## 8 Conclusions

This paper focused on providing theoretical guarantees on the reliability and safety of kernel-based algorithms for machine learning and system identification. Such procedures are increasingly being deployed in safety-critical applications, including autonomous driving and smart healthcare. Complementing the estimates with reliable confidence measures is therefore crucial in these settings, as failures could be catastrophic. Current approaches are subject to important limitations connected with the excessive level of conservatism of the uncertainty regions they return. Modern applications require instead bounds that are exact for a wide class of useful models. The uncertainty regions established in this paper have exact coverage probability under a minimal set of assumptions on the distributions of the noises, and are computationally tractable. Thus, they can be employed to efficiently quantify the reliability of many data-driven predictions and guide machine learning algorithms towards safe decisions. Applications abound, including regression, sparse estimation, and reinforcement learning.

Our bounds have been obtained by designing a novel Bayesian frequentist framework that deeply expands the original SPS approach introduced in [14]. Many other developments of this work are however still possible in several directions, e.g., by combining BFB with undermodelling detection mechanisms [12] and by extending the BFB approach to the case of non-exogenous regressors where the regression matrix may depend on past outputs [60,10]. This will provide a basis to obtain exact bounds for more complex systems. Further, it would be interesting to extend the proposed framework so as to exploit information on higher order moments (beyond first and second moments) as partial information on the distribution of the noises. Other applications could then include on-line reinforcement learning, where uncertainty has to be computed in real-time as new training data become available. Finally, evaluating the performance of our bounds in real testbeds is another compelling direction of research that we plan to investigate in the future.

*Data availability.* The MATLAB code which implements all the algorithms described in this paper can be found at https://www.dei.unipd.it/∼giapi/software.html.

## References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.

[2] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice-Hall, Englewood Cliffs, N.J., USA, 1979.

[3] C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342, 2010.

[4] A. Aravkin, B.M. Bell, J.V. Burke, and G. Pillonetto. An $\ell_1$-Laplace robust Kalman smoother. *IEEE Transactions on Automatic Control*, 56(12):2898–2911, 2011.

[5] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.

[6] M. J. Bayarri and J. O. Berger. The interplay of Bayesian and Frequentist analysis. *Statistical Sciences*, 19(1):58–80, 2004.

[7] G. Blanchard and N. Mücke. Optimal rates for regularization of statistical inverse learning problems. *Foundations of Computational Mathematics*, 18:971–1013, 2018.

[8] R. Boczar, N. Matni, and B. Recht. Finite-data performance guarantees for the output-feedback control of an unknown system. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2994–2999, 2018.

[9] M. C. Campi and E. Weyer. Guaranteed non-asymptotic confidence regions in system identification. *Automatica*, 41(10):1751–1764, 2005.

[10] A. Carè, B. Cs. Csáji, M. C. Campi, and E. Weyer. Finite-sample system identification: An overview and a new correlation method. *IEEE Control Systems Letters*, 2(1):61–66, 2018.

[11] A. Carè, G. Pillonetto, and M. C. Campi. Uncertainty bounds for kernel-based regression: A Bayesian SPS approach. In *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2018.

[12] A. Car, M.C. Campi, B.Cs. Csji, and E. Weyer. Facing undermodelling in Sign-Perturbed-Sums system identification. *Systems & Control Letters*, 153:104936, 2021.

[13] B. Cs. Csáji. Non-asymptotic confidence regions for regularized linear regression estimates. In István Faragó, Ferenc Izsák, and Péter L. Simon, editors, *Progress in Industrial Mathematics at ECMI 2018*, pages 605–611, Cham, 2019. Springer International Publishing.

[14] B. Cs. Csáji, M. C. Campi, and E. Weyer. Sign-perturbed sums: A new system identification approach for constructing exact non-asymptotic confidence regions in linear regression models. *IEEE Transactions on Signal Processing*, 63(1):169–181, 2015.

[15] B. Cs. Csáji and K. B. Kis. Distribution-free uncertainty quantification for kernel methods by gradient perturbations. *Machine Learning*, 108(8):1677–1699, 2019.

[16] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, 20(4):633–679, 2020.

[17] M.P. Deisenroth, D. Fox, and C.E. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):408–423, 2015.

[18] J. Dezert and C. Musso. An efficient method for generating points uniformly distributed in hyperellipsoids. In *Proceedings of the Workshop on Estimation, Tracking and Fusion, Naval Postgraduate School in Monterey*, 2011.

[19] H. Drucker, C.J.C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In *Advances in Neural Information Processing Systems*, volume 9, pages 155–161, 1997.

[20] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in Computational Mathematics*, 13:1–50, 2000.

[21] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360, 2001.

[22] R. Frigola, F. Lindsten, T.B. Schön, and C.E. Rasmussen. Bayesian inference and learning in Gaussian process state-space models with particle MCMC. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

[23] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, 2nd edition, 2004.

[24] W. Gilks, S. Richardson, and D. Spiegehalter. *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall, 1996.

[25] A. Girard, C. E. Rasmussen, J. Quiñonero-Candela, and R. Murray-Smith. Bayesian regression and Gaussian process priors with uncertain inputs - application to multiple-step ahead time series forecasting. In *Proceedings of Neural Information Processing Systems (NIPS) conference*, 2003.

[26] F. Girosi. An equivalence between sparse approximation and support vector machines. Technical report, Cambridge, MA, USA, 1997.

[27] F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7(2):219–269, 1995.

[28] P. Goldberg, C. Williams, and C. Bishop. Regression with input-dependent noise: A Gaussian process treatment. In *Advances in Neural Information Processing Systems (NIPS)*, volume 10, 1998.

[29] A. Goldenshluger. Nonparametric estimation of transfer functions: rates of convergence and adaptation. *IEEE Transactions on Information Theory*, 44(2):644–658, 1998.

[30] G. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[31] Z.C. Guo and D.X. Zhou. Concentration estimates for learning with unbounded sampling. *Adv. Comput. Math.*, 38(1):207–223, 2013.

[32] T. J. Hastie, R. J. Tibshirani, and J. Friedman. *The Elements of Statistical Learning. Data Mining, Inference and Prediction*. Springer, Canada, 2001.

[33] M. Heinonen, H. Mannerstram, J. Rousu, S. Kaski, and H. Lahdesmaki. Non-stationary Gaussian process regression with Hamiltonian Monte Carlo. In A. Gretton and C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 732–740, Cadiz, Spain, 2016.

[34] G. A. Hewer, R. D. Martin, and J. Zeh. Robust preprocessing for Kalman filtering of glint noise. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23(1):120–128, 1987.

[35] P. J. Huber. *Robust Statistics*. John Wiley and Sons, New York, NY, USA, 1981.

[36] J. Johndrow, P. Orenstein, and A. Bhattacharya. Scalable approximate mcmc algorithms for the horseshoe prior. *Journal of Machine Learning Research*, 21(73):1–61, 2020.

[37] R.E. Kass and A.E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90:773–795, 1995.

[38] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard. Most likely heteroscedastic Gaussian process regression. In *Proceedings of the 24th International Conference on Machine Learning (ICML 2007)*, 2007.

[39] G. Kimeldorf and G. Wahba. A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502, 1970.

[40] M. Lazaro-Gredilla and M. Titsias. Variational heteroscedastic Gaussian process regression. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 841–848, 2011.

[41] Q.V. Le, A. Smola, and S. Canu. Heteroscedastic Gaussian process regression. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pages 489–496, 2005.

[42] P. Magni, R. Bellazzi, and G. De Nicolao. Bayesian function learning using MCMC methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(12):1319–1331, 1998.

[43] C. A. Micchelli, Y. Xu, and H. Zhang. Universal kernels. *Journal of Machine Learning Research*, 7:2651–2667, 2006.

[44] T. Park and G. Casella. The Bayesian Lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

[45] G. Pillonetto, A. Carè, and M. C. Campi. Kernel-based SPS. *IFAC-PapersOnLine*, 51(15):31 – 36, 2018. 18th IFAC Symposium on System Identification (SYSID).

[46] G. Pillonetto and G. De Nicolao. A new kernel-based approach for linear system identification. *Automatica*, 46(1):81–93, 2010.

[47] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung. Kernel methods in system identification, machine learning and function estimation: a survey. *Automatica*, 50(3):657–682, 2014.

[48] T. Poggio and F. Girosi. Networks for approximation and learning. In *Proceedings of the IEEE*, volume 78, pages 1481–1497, 1990.

[49] N.G. Polson, J.G. Scott, and J. Windle. The Bayesian bridge. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(4):713–733, 2014.

[50] G. Prando, D. Romeres, G. Pillonetto, and A. Chiuso. Classical vs. Bayesian methods for linear system identification: Point estimators and confidence sets. In *2016 European Control Conference (ECC)*, pages 1365–1370, 2016.

[51] A. Raftery and S. Lewis. Implementing mcmc. In S. Richardson W. Gilks and D. Spiegehalter, editors, *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall, 1996.

[52] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[53] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning. MIT Press, 2001.

[54] S. Smale and D.X. Zhou. Learning theory estimates via integral operators and their approximations. *Constructive Approximation*, 26:153–172, 2007.

[55] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[56] R. Tibshirani. Regression shrinkage and selection via the LASSO. *Journal of the Royal Statistical Society, Series B.*, 58:267–288, 1996.

[57] S. Tu, R. Boczar, A. Packard, and B. Recht. Non-asymptotic analysis of robust control from coarse-grained identification. arXiv, 1707.04791, 2017.

[58] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, NY, USA, 1998.

[59] V. Volpe. *Identification of dynamical systems with finitely many data points*. University of Brescia, M. Sc. Thesis, 2015.

[60] V. Volpe, B. Cs. Csáji, A. Carè, E. Weyer, and M. C. Campi. Sign-Perturbed Sums (SPS) with instrumental variables for the identification of ARX systems. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 2115–2120, 2015.

[61] G. Wahba. *Spline models for observational data*. SIAM, Philadelphia, 1990.

[62] C. Wang and D.X. Zhou. Optimal learning rates for least squares regularized regression with unbounded sampling. *Journal of Complexity*, 27(1):55–67, 2011.

[63] E. Weyer, M. Campi, and B.C. Csaji. Asymptotic properties of SPS confidence regions. *Automatica*, 82:287 – 294, 2017.

[64] Q. Wu, Y Ying, and D.X. Zhou. Learning rates of least-square regularized regression. *Foundations of Computational Mathematics*, 6:171–192, 2006.

[65] M. Yuan and T. Tony Cai. A reproducing kernel Hilbert space approach to functional linear regression. *Annals of Statistics*, 38:3412–3444, 2010.

[66] F. Zhang. *Matrix Theory: Basic Results and Techniques*. Springer, 2011.

[67] P. Zhao and B. Yu. On model selection consistency of LASSO. *Journal of Machine Learning Research*, 7:2541–2563, 2006.

## 9 Appendix

### *9.1 From SPS to BFB through a toy example*

This section expands the informal introduction to SPS that motivated the new BFB in Section 3. There, we have seen that classic SPS approaches build confidence regions around classic and regularized least squares (e.g., [14] and [15], respectively), but treat $\theta^0$ as a deterministic quantity. The goal of this Section is to highlight the benefit of treating $\theta^0$ as a random variable, as done in the proposed BFB. The first Example aligns with the original (regularized) SPS set-up, where randomization affects noise components only.

**Example 1** *Assume that we only have one observation $y_1 = \theta_1 + \nu_1$ and our prior is the generation mechanism (3) with $\mu = 0$, $\omega$ distributed as a zero-mean unit-variance Gaussian, and $C = c$, $c \in \mathbb{R}$. A 50%-probability region is constructed as $R = \{\theta : \|\tilde{H}_0(\theta)\| \leq \|\tilde{H}_1(\theta)\|\} = \{\theta : ((y-\theta) - \frac{1}{c^2}\theta)^2 \leq (\pm(y-\theta) - \frac{1}{c^2}\theta)^2\}$, where $\tilde{H}_i(\theta)$ has been defined in (11). Note that in this case the classic least-squares oriented SPS region is obtained when $c = \infty$ (no prior), which always gives an uninformative region. However, no matter how small $c$ is (i.e., how strong the prior is), the region remains uninformative every time that $\pm$ is $+$ (which happens with probability 0.5). The reason is that the region $R$ is guaranteed to include $\theta^0$ independently of how $\theta^0$ is generated: the probability of including $\theta^0$ is the same conditionally to any value of $\theta^0$, no matter if this value is absolutely unlikely given the prior adopted by the user.*

We show in the next Example that BFB, which involves random perturbation also for $\theta^0$, is capable of returning informative regions in the context above.

**Example 2** *Consider the same set-up as in Example 1. Let $\omega$ be a random variable from a zero-mean, unit-variance distribution $\mathcal{D}$ from which we can generate samples. A region that is guaranteed to include $\theta^0$ with probability 0.5 with respect to the variability of $\nu_1$ and $\theta^0 = c\omega$ is $\{\theta : ((y-\theta) - \frac{1}{c^2}\theta)^2 \leq (\pm(y-\theta) - \frac{1}{c^2}\tilde{\omega})^2\}$, where $\tilde{\omega}$ is an independent sample from the distribution $\mathcal{D}$. The fact that the quadratic term in $\theta$ grows faster at the left-hand side than at the right-hand side of the inequality reveals immediately that the region is always limited (except for the case $c = \infty$).*

### *9.2 Proof of Theorem 1*

In this Section we prove Theorem 1 and explain the rationale of Algorithm 1. To this purpose, we will state

first some instrumental definitions and lemmas.

**Definition 1 (Exchangeable sequence)** *Let* $\mathbf{x} := (x_1, \dots, x_n)$ *be a finite sequence of random variables. The sequence* $\mathbf{x}$ *is said to be* exchangeable *if, for every permutation* $\sigma$ *of the indices* $\{1, \dots, n\}$*, it holds* $\mathbf{x} \overset{d}{=} \mathbf{x}_\sigma$*, where* $\mathbf{x}_\sigma := (x_{\sigma(1)}, \dots, x_{\sigma(n)})$ *and* $\overset{d}{=}$ *denotes equality in distribution.*

**Lemma 1** *Let* $\mathbf{x} := (x_1, \dots, x_n)$ *be an exchangeable sequence of real-valued random variables, and let* $\pi$ *be a uniformly distributed random permutation of the set* $\{1, \dots, n\}$*. Further, let* $\mathcal{R}(\mathbf{x})$ *be the function that, given the values of* $x_1, \dots, x_n$*, returns the ranking of* $x_1$ *in the ascending-ordered sequence of all* $x_1, \dots, x_n$*. In case there exist indices* $i \in \{1, \dots, n\}$ *such that* $x_i = x_1$*, we assume that* $x_1$ *precedes* $x_i$ *in the ranking if and only if* $\pi(1) < \pi(i)$*. Then,* $\mathcal{R}(\mathbf{x})$ *is uniformly distributed over* $\{1, \dots, n\}$*.*

**Proof.** Since $\mathbf{x}$ is exchangeable by assumption, all possible permutations of the elements of $\mathbf{x}$ are equally probable. Thus, for the values of $\mathbf{x}$ which do not feature ties between $x_1$ and other elements of the sequence, $\mathcal{R}(\mathbf{x})$ takes values in $\{1, \dots, n\}$ with equal probability. In case there exist one or more indices $i \in \{1, \dots, n\}$ such that $x_i = x_1$, then the random permutation $\pi$, which is uniformly distributed over all permutations of $\{1, \dots, n\}$, is such that $\mathcal{R}(\mathbf{x})$ takes equal probability over all admissible rankings of $x_1$. Hence, we conclude that $\mathcal{R}(\mathbf{x})$ is uniformly distributed over $\{1, \dots, n\}$. □

**Lemma 2** *Let* $D \in \mathbb{R}^{n \times n}$ *be a diagonal matrix with* 0*,* 1*,* $-1$ *diagonal entries. Then,*

$$H := \begin{bmatrix} I_n & D \\ D & I_n \end{bmatrix} \succeq 0, \tag{20}$$

*where* $\succeq$ *denotes the Löwner partial ordering of symmetric matrices.*

**Proof.** Define the matrix $T := \left[\begin{smallmatrix} I_n & -D \\ 0 & I_n \end{smallmatrix}\right]$, which is partitioned conformably to $H$. It holds

$$H \succeq 0 \iff H' := THT^\top = \begin{bmatrix} I_n - D^2 & 0 \\ 0 & I_n \end{bmatrix} \succeq 0. \tag{21}$$

By definition of $D$, it follows that $I_n - D^2$ is a diagonal matrix with either 0 or 1 diagonal entries. Thus, from (21), $H' \succeq 0$, which concludes the proof. □

**Proof of Theorem 1.** The proof is divided in two parts. First, we prove that the Bayesian frequentist region $\Theta(y)$ obtained via Algorithm 1 contains $\theta^0$ with the exact coverage level, according to equation (16); then we show that $\Theta(y)$ is the union of a finite number of convex sets and, if non-empty, always contains the estimate (4).

As for the first part, given a random vector $\theta \in \mathbb{R}^m$ and resorting to (12) and (13), we define the prediction errors

$$\rho_t(\theta) := z_t - \Omega(t,:)\theta, \quad t = 1, \dots, N, \tag{22}$$

$$\rho_t^i(\theta) := \xi_t^i(z_t - \Omega(t,:)\theta) + \zeta_t^i, \quad t = 1, \dots, N, \;\; i = 1, \dots, r-1. \tag{23}$$

Further, recalling that $R_N = \Omega^\top \Omega / N$, we define the test functions

$$S_0(\theta) := R_N^{-1/2} \frac{1}{N} \sum_{t=1}^{N} \Omega(t,:)^\top \rho_t(\theta) \tag{24}$$

$$S_i(\theta) := R_N^{-1/2} \frac{1}{N} \sum_{t=1}^{N} \Omega(t,:)^\top \rho_t^i(\theta), \quad i = 1, \dots, r-1 \tag{25}$$

generalizing the SPS approach.
Notice that $\|S_0(\theta)\|^2$ and $\|S_i(\theta)\|^2$ are quadratic functions of $\theta$ which generalize the SPS approach in line with the discussion of Section 3 (when $B = I_n$, (25) is precisely (11), pre-multiplied by the shaping matrix $(A^\top A)^{-\frac{1}{2}}$ rescaled by $1/\sqrt{N}$), and that can be written, after some simple computations, as

$$\|S_0(\theta)\|^2 = (\theta - \hat{\theta})^\top R_N (\theta - \hat{\theta}), \tag{26}$$

$$\|S_i(\theta)\|^2 = \theta^\top Q_i R_N^{-1} Q_i \theta - 2\theta^\top Q_i R_N^{-1} \psi_i + \psi_i^\top R_N^{-1} \psi_i, \tag{27}$$

where $Q_i$ and $\psi_i$, $i = 1, \dots, r-1$, are as in Algorithm 1 and $\hat{\theta}$ is the estimate (4). Using (26) and (27), we can rewrite the sets (14) in Algorithm 1 as

$$\mathcal{E}_i = \{\theta \in \mathbb{R}^m \;:\; \|S_0(\theta)\|^2 \leq_{\pi,i} \|S_i(\theta)\|^2\}, \quad i = 1, \dots, r-1, \tag{28}$$

where we recall that the ordering "$\leq_{\pi,i}$" is defined as "$\leq$" if $\pi(0) < \pi(i)$, and as "$<$" otherwise. Finally, we define the "ranking" function

$$\mathcal{R}(\theta) := r - \sum_{i=1}^{r-1} \mathbf{1}(\|S_0(\theta)\| \leq_{\pi,i} \|S_i(\theta)\|), \tag{29}$$

where $\mathbf{1}(\cdot)$ stands for the indicator function which equals 1 when the argument is true, and 0 otherwise. The function $\mathcal{R}(\theta)$ in (29) equals the ranking of $\|S_0(\theta)\|$ in the ascending-ordered sequence of all $\|S_i(\theta)\|$, $i = 0, \dots, r-1$. Note, in particular, that if there exists an index $i$ such that $\|S_0(\theta)\| = \|S_i(\theta)\|$, then $\|S_0(\theta)\|$ precedes $\|S_i(\theta)\|$ in the ranking if and only if $\pi(0) < \pi(i)$.

By definition, $\Theta(y) = \bigcup_{i=1}^{r-1} \mathcal{E}_i$, so that, in view of (28) and (29),

$$\mathbb{P}(\theta^0 \in \Theta(y)) = \mathbb{P}(\mathcal{R}(\theta^0) \leq r-1). \tag{30}$$

This, in turn, implies that (16) holds true if and only if

$$\mathbb{P}(\mathcal{R}(\theta^0) = r) = 1/r. \tag{31}$$

We will prove (31) by showing that $\mathcal{R}(\theta^0)$ is uniformly distributed over $\{1,\dots,r\}$. To this end, using that $\nu_t = B^{-1}(t,:)(y - A\theta^0)$ (see (2)) and $\omega_k = C^{-1}(k,:)(\mu - \theta^0)$ (see (3)), we first rewrite the function $S_0(\theta)$ in (24) evaluated at $\theta = \theta^0$ as

$$S_0(\theta^0) = R_N^{-1/2} \frac{1}{N} \left[ \sum_{t=1}^{n} \Omega(t,:)^\top \nu_t - \sum_{k=1}^{m} C^{-1}(k,:)^\top \omega_k \right]. \tag{32}$$

Note that $S_0(\theta^0)$ is a random variable through its dependence on $\nu$ and $\omega$. We define

$$Z(\mathbf{r}) := \|S_0(\theta^0)\| \tag{33}$$

as the deterministic function that, given the values of $\mathbf{r} := (\nu_1,\dots,\nu_n,\omega_1,\dots,\omega_m)$, computes $\|S_0(\theta^0)\|$ according to (32). Further, for $i = 1,\dots,r-1$, we define $\boldsymbol{\xi}_i$ and $\boldsymbol{\zeta}_i$, $i = 1,\dots,r-1$, as the sequences $(\xi_1^i,\dots,\xi_N^i)$ and $(\zeta_1^i,\dots,\zeta_N^i)$, respectively. Then, $\|S_i(\theta^0)\|$ can be compactly rewritten as

$$\|S_i(\theta^0)\| = Z(\boldsymbol{\xi}_i \circ \mathbf{r} + \boldsymbol{\zeta}_i), \quad i = 1,\dots,r-1, \tag{34}$$

where $Z(\cdot)$ is the same function as before and $\circ$ denotes component-wise multiplication of sequences. Next, we define $\mathbf{r}' := \boldsymbol{\xi}_0 \circ \mathbf{r} + \boldsymbol{\zeta}_0$, where $\boldsymbol{\xi}_0$ is a new independent sample of the sequence of $\{\xi_t^i\}$, defined according to (13), and $\boldsymbol{\zeta}_0$ is a new independent sample of the sequence of $\{\zeta_t^i\}$, and

$$\mathcal{R}'(\theta^0) := r - \sum_{i=1}^{r-1} \mathbf{1}(Z(\mathbf{r}') \leq_{\pi,i} Z(\boldsymbol{\xi}_i \circ \mathbf{r}' + \boldsymbol{\zeta}_i)). \tag{35}$$

Notice that, since the distribution of the components of the independent noises in $\mathbf{r}$ is symmetric around zero and the elements of $\boldsymbol{\xi}_0$ are either i.i.d. random signs or zero with probability one, we have $\mathbf{r}' \stackrel{d}{=} \mathbf{r}$, where the symbol $\stackrel{d}{=}$ denotes equality in distribution. It follows that

$$\begin{aligned} &\Big(Z(\mathbf{r}'), Z(\boldsymbol{\xi}_1 \circ \mathbf{r}' + \boldsymbol{\zeta}_1), \dots, Z(\boldsymbol{\xi}_N \circ \mathbf{r}' + \boldsymbol{\zeta}_N)\Big) \qquad (36)\\ &\stackrel{d}{=} \Big(Z(\mathbf{r}), Z(\boldsymbol{\xi}_1 \circ \mathbf{r} + \boldsymbol{\zeta}_1), \dots, Z(\boldsymbol{\xi}_N \circ \mathbf{r} + \boldsymbol{\zeta}_N)\Big), \end{aligned}$$

and, therefore,

$$\mathcal{R}(\theta^0) \stackrel{d}{=} \mathcal{R}'(\theta^0). \tag{37}$$

Conditioning on a given value of $\mathbf{r}$, say $\bar{\mathbf{r}}$, we can write

$$\begin{aligned} &\Big(Z(\mathbf{r}'), Z(\boldsymbol{\xi}_1 \circ \mathbf{r}' + \boldsymbol{\zeta}_1), \dots, Z(\boldsymbol{\xi}_N \circ \mathbf{r}' + \boldsymbol{\zeta}_N)\Big) \qquad (38)\\ =&\Big(Z(\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0), Z(\boldsymbol{\xi}_1 \circ (\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0) + \boldsymbol{\zeta}_1), \dots\\ &\qquad \dots, Z(\boldsymbol{\xi}_N \circ (\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0) + \boldsymbol{\zeta}_N)\Big)\\ =&\Big(Z(\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0), Z((\boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_1) \circ \bar{\mathbf{r}} + \boldsymbol{\xi}_1 \circ \boldsymbol{\zeta}_0 + \boldsymbol{\zeta}_1), \dots\\ &\qquad \dots, Z((\boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_N) \circ \bar{\mathbf{r}} + \boldsymbol{\xi}_N \circ \boldsymbol{\zeta}_0 + \boldsymbol{\zeta}_N)\Big). \end{aligned}$$

Since $\boldsymbol{\xi}_0, \boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_N$ are i.i.d. sequences, independent of $\boldsymbol{\zeta}_0,\dots,\boldsymbol{\zeta}_N$, and, for all $i$, $\boldsymbol{\xi}_i \circ \boldsymbol{\zeta}_0$ equals the sequence of all zeros with probability one, it follows that the sequence $(Z(\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0), Z((\boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_1) \circ \bar{\mathbf{r}} + \boldsymbol{\xi}_1 \circ \boldsymbol{\zeta}_0 + \boldsymbol{\zeta}_1), \dots, Z((\boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_N) \circ \bar{\mathbf{r}} + \boldsymbol{\xi}_N \circ \boldsymbol{\zeta}_0 + \boldsymbol{\zeta}_N)$ is exchangeable. By Lemma 1, this implies that the ranking $\mathcal{R}'(\theta^0)$ of $Z(\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0)$ in the ascending-ordered (with respect to "$\leq_{\pi,i}$") sequence $(Z(\boldsymbol{\xi}_0 \circ \bar{\mathbf{r}} + \boldsymbol{\zeta}_0), Z((\boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_1) \circ \bar{\mathbf{r}} + \boldsymbol{\xi}_1 \circ \boldsymbol{\zeta}_0 + \boldsymbol{\zeta}_1), \dots, Z((\boldsymbol{\xi}_0 \circ \boldsymbol{\xi}_N) \circ \bar{\mathbf{r}} + \boldsymbol{\xi}_N \circ \boldsymbol{\zeta}_0 + \boldsymbol{\zeta}_N)$ is uniformly distributed over $\{1,\dots,r\}$. Thus, we have proved that $\mathbb{P}(\mathcal{R}'(\theta^0) = r \mid \mathbf{r} = \bar{\mathbf{r}}) = 1/r$, irrespective of the choice of $\bar{\mathbf{r}}$; by integrating over $\bar{\mathbf{r}}$, we conclude that

$$\mathbb{P}(\mathcal{R}'(\theta^0) = r) = 1/r, \tag{39}$$

and (31) follows from (37). This concludes the first part of the proof.

As for the second part, we will prove that each set $\mathcal{E}_i$, $i = 1,\dots,r-1$, is convex and contains (4) if non-empty. This part of the proof is based on the same arguments of [14, Appendix B], which are adapted to the present context as follows. In view of (28), proving convexity of $\mathcal{E}_i$ is equivalent to showing that the quadratic function $\|S_0(\theta)\|^2 - \|S_i(\theta)\|^2$ is convex. Thus, it suffices to prove that each $A_i = R_N - Q_i R_N^{-1} Q_i$ is a positive semidefinite matrix, or, equivalently, that

$$R_N \succeq Q_i R_N^{-1} Q_i \tag{40}$$

where $\succeq$ denotes the Löwner partial ordering of symmetric matrices. Since $R_N$ is positive definite, using a Schur complement argument (see, e.g., [66, Section 7.3]), we have that (40) holds if and only if

$$H_i := \begin{bmatrix} R_N & Q_i \\ Q_i & R_N \end{bmatrix} \succeq 0. \tag{41}$$

Next, notice that, using the definition of $\Omega$ in (12), $R_N$ and $Q_i$ can be rewritten in matrix form as

$$R_N = \frac{1}{N}(B^{-1}A)^\top B^{-1}A + \frac{1}{N}(C^{-1})^\top C^{-1} \tag{42}$$

$$Q_i = \frac{1}{N}(B^{-1}A)^\top D_n^i B^{-1}A + \frac{1}{N}(C^{-1})^\top D_m^i C^{-1}, \tag{43}$$

with $D_n^i := \text{diag}(\xi_1^i, \ldots, \xi_n^i)$ and $D_m^i := \text{diag}(\xi_{n+1}^i, \ldots, \xi_{n+m}^i)$, where $\text{diag}(x_1, \ldots, x_n)$ denotes a diagonal matrix with entries $x_1, \ldots, x_n$ on the diagonal. Therefore, $H_i$ in (41) can be written as

$$H_i = \frac{1}{N} H_n^i + \frac{1}{N} H_m^i, \tag{44}$$

where

$$H_n^i := \begin{bmatrix} B^{-1}A & 0 \\ 0 & B^{-1}A \end{bmatrix}^\top \begin{bmatrix} I_n & D_n^i \\ D_n^i & I_n \end{bmatrix} \begin{bmatrix} B^{-1}A & 0 \\ 0 & B^{-1}A \end{bmatrix}, \tag{45}$$

$$H_m^i := \begin{bmatrix} C^{-1} & 0 \\ 0 & C^{-1} \end{bmatrix}^\top \begin{bmatrix} I_m & D_m^i \\ D_m^i & I_m \end{bmatrix} \begin{bmatrix} C^{-1} & 0 \\ 0 & C^{-1} \end{bmatrix}. \tag{46}$$

Since, by Lemma 2, the middle matrices of $H_n^i$ and $H_m^i$ are always positive semidefinite, the matrices $H_n^i$ and $H_m^i$, and hence $H_i$, are positive semidefinite. Hence, from (40) we conclude that each $A_i$ is positive semidefinite, so that each set $\mathcal{E}_i$, $i = 1, \ldots, r-1$, is convex. To prove that the estimate $\hat{\theta}$ in (4) is contained in the Bayesian frequentist region $\Theta(y)$, if this region is non-empty, we show that each set $\mathcal{E}_i$, $i = 1, \ldots, r-1$ contains $\hat{\theta}$ if non-empty. To this end, we distinguish two cases which depend on the random permutation $\pi$:

(1) $\underline{\pi(0) < \pi(i)}$: In this case, the set $\mathcal{E}_i$ is

$$\mathcal{E}_i = \{\theta \in \mathbb{R}^m \ : \ \|S_0(\theta)\|^2 \leq \|S_i(\theta)\|^2\}. \tag{47}$$

From the definition of $\|S_0(\theta)\|^2$ in (26), it holds $\|S_0(\hat{\theta})\|^2 = 0$, which implies $\hat{\theta} \in \mathcal{E}_i$.

(2) $\underline{\pi(0) > \pi(i)}$: In this case, the set $\mathcal{E}_i$ is

$$\mathcal{E}_i = \{\theta \in \mathbb{R}^m \ : \ \|S_0(\theta)\|^2 < \|S_i(\theta)\|^2\}. \tag{48}$$

and can be can be either empty or not. If $\mathcal{E}_i$ is non-empty, then $\hat{\theta} \in \mathcal{E}_i$, as we prove next. Assume, for the sake of contradiction, that $\hat{\theta} \notin \mathcal{E}_i$. Since $\mathcal{E}_i$ is non-empty, there exists $\bar{\theta} \neq \hat{\theta}$ such that $\bar{\theta} \in \mathcal{E}_i$. Such $\bar{\theta}$ satisfies $\|S_0(\bar{\theta})\|^2 < \|S_i(\bar{\theta})\|^2$. Further, from (26) and (48), it must be $\|S_0(\hat{\theta})\|^2 = \|S_i(\hat{\theta})\|^2 = 0$ which implies that both $\|S_0(\theta)\|^2$ and $\|S_i(\theta)\|^2$ have a minimum at $\hat{\theta}$ with value 0, and therefore the gradient of $\|S_i(\theta)\|^2 - \|S_0(\theta)\|^2$ is zero at $\hat{\theta}$. Thus, over the segment $\theta_\lambda = (1-\lambda)\hat{\theta} + \lambda\bar{\theta}$, $\lambda \in [0, 1]$, the function $\|S_i(\theta_\lambda)\|^2 - \|S_0(\theta_\lambda)\|^2$ grows from 0 with zero derivative to a strictly positive value and therefore $\|S_i(\theta_\lambda)\|^2 - \|S_0(\theta_\lambda)\|^2$ must have a positive second-order derivative for some $\lambda \in (0, 1]$. This contradicts the fact that $\|S_0(\theta)\|^2 - \|S_i(\theta)\|^2$ has a positive semidefinite Hessian (cf. equation (40)), yielding the desired conclusion.

This concludes the second part and completes the proof. □

### *9.3 Proof of Theorem 2*

The proof proceeds along the lines of the one of Theorem 1 presented in Section 9.2, except that now in the summation defining the functions $S_i(\theta)$ in (25) for $i = 1, \ldots, r-1$, the first $\bar{n}$ and the last $N - n$ terms are equal to the ones of $S_0(\theta)$. This fact is key since it ensures that (31) can be replaced by $\mathbb{P}(\mathcal{R}(\theta^0) = r \mid \bar{y}, \theta^0) = 1/r$, and the inclusion property holds conditionally on any realization of $\bar{y}$ and $\theta^0$. The rest of the proof then remains unchanged. □

### *9.4 Invertibility of $A_i$*

Consider the uncertainty region $\Theta(y)$ returned by Algorithm 1, which is defined as the union of convex sets $\mathcal{E}_i$, $i = 1, \ldots, r-1$, in (14). The sets $\mathcal{E}_i$ can be unbounded in general. In practice, however, the probability for $\mathcal{E}_i$ to be unbounded is negligible, as we illustrate next.

From the definition of $\mathcal{E}_i$ in (14), if the set $\mathcal{E}_i$ is unbounded then the matrix $A_i$ is singular. Indeed, if $A_i$ is invertible, then $A_i$ is positive definite and $\mathcal{E}_i$ coincides with the ellipsoid

$$\mathcal{E}_i = \{\theta \in \mathbb{R}^m \ : \ (\theta - \bar{b}_i)^\top A_i (\theta - \bar{b}_i) \leq_{\pi,i} \bar{c}_i\}, \tag{49}$$

where $\bar{b}_i = -A_i^{-1} b_i$ and $\bar{c}_i = -c_i + b_i^\top A_i^{-1} b_i$, which is always bounded. Matrix $A_i$ is singular if and only if $H_n^i + H_m^i$ is singular, where $H_n^i$ and $H_m^i$ are the positive semidefinite matrices defined in (45) and (46) which depend on the random variables $\xi_1^i, \ldots, \xi_N^i$ (see the second part of the proof of Theorem 1). Thus, $A_i$ is singular if and only if the null spaces of $H_n^i$ and $H_m^i$ have a non-empty intersection. This event may have a positive probability. For instance, if we perturb both $\nu$ and $\theta^0$ with random signs (that is, all components of $\epsilon$ belong to set $\mathcal{B}$, thus all $\xi_1^i, \ldots, \xi_N^i$ take values $\pm 1$ with equal probability), then $A_i$ is singular when all $\{\xi_j^i\}$ have the same sign. However, such probability becomes negligible when the number of measurements in $y$ is larger than the dimension of the unknown vector $\theta^0$ and/or when the distributions of some components of the noises $\nu$ and $\omega$ are assumed to be known. Remarkably, Algorithm 1 returns an informative region even if $n \leq m$, provided that the distribution of $\omega$ is known. In Table 1 we numerically evaluate the probability of $A_i$ to be singular for $m = 3$, randomly generated matrices $A$, $B$, $C$ and different values of both the number of measurements $n$ and of the number of noise components with known distribution.

### *9.5 Comparison with Markov Chain Monte Carlo*

In this section we compare the bounds obtained via Algorithm 1 with Bayes intervals computed via MCMC. The

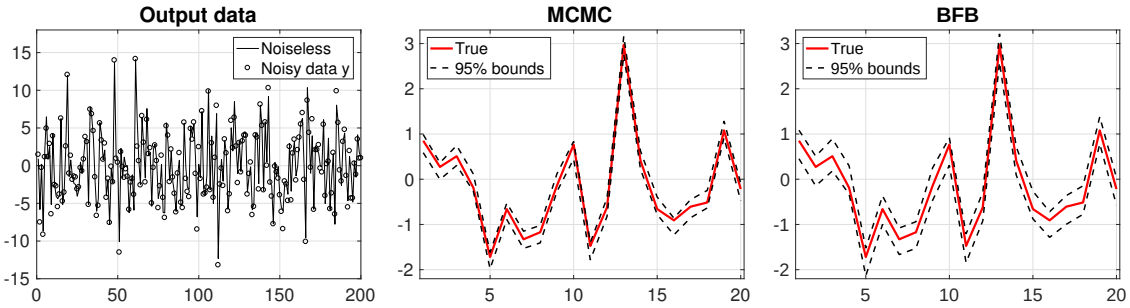


Fig. 5. *Laplacian noise* Output data (left), realization of $\theta^0$ (solid line, middle and right) and 95% confidence intervals returned by MCMC (middle) and the new BFB (right).

Table 1
Probability of $A_i$ to be singular for $m = 3$ and matrices $A$, $B$, $C$ with i.i.d. normal entries. The indices $t$ of the $\epsilon_t \in \mathcal{A}$ have been drawn randomly from the set $\{1, \dots, N\}$.

| | $\#\ \epsilon_t \in \mathcal{A}$ | | |
|---|---|---|---|
| | 0 | 1 | 2 |
| $n = 8$ | 6.54% | 2.15% | 0.39% |
| $n = 10$ | 2.25% | 0.63% | 0.0976% |
| $n = 12$ | 0.74% | 0.18% | 0.0244% |
| $n = 14$ | 0.23% | 0.0518% | 0.0061% |
| $n = 16$ | 0.0728% | 0.0145% | 0.00152% |
| $n = 18$ | 0.0221% | 0.00401% | 0.00038% |
| $n = 20$ | 0.0066% | 0.00109% | 0.00009% |

comparison requires a *caveat*: both approaches provide non-asymptotic and exact uncertainty bounds around the (regularized) estimate of $\theta^0$, but the two regions have different interpretations. As regards BFB, the probability (16) in Theorem 1 holds unconditionally from $\theta^0$ and $y$. On the other hand, Bayes intervals leverage the posterior distribution

$$\mathbf{p}(\theta^0|y) = \frac{\mathbf{p}(y|\theta^0)\mathbf{p}(\theta^0)}{\mathbf{p}(y)} \tag{50}$$

and compute the region with coverage level $\alpha$ by extracting the suitable quantiles of this conditional probability. Therefore, such regions are built conditionally on the value of $y$, and the $\alpha$-level guarantee holds for that specific realization. Thus, when comparing the bounds for a single experiment, we do not discuss whether the bounds contain the realization of $\theta^0$ or not, but we focus on their size.

Bayes regions are known to perform very well in terms of tightness, but they could be difficult to obtain: the posterior in (50) is often not available in closed form, so the integrals required to compute the $\alpha$-level region are analytically intractable. In those cases, stochastic simulation schemes are widely used. In particular, MCMC obtains $\mathbf{p}(\theta^0|y)$ in sampled form by constructing a Markov chain whose invariant distribution is the posterior of interest. This may be far from trivial, since a careful choice of proposal distributions has to be done; moreover, a large number of samples has to be drawn to reach stationarity and to extract the desired quantiles.

We now consider the set-up in the first kind of experiments of Section 5 (i.e., the measurements model with Gaussian prior on $\theta^0$ and Laplacian noise $\nu$) and show that BFB yields regions that are comparable to the ones of MCMC in terms of tightness, but require less computational time. The same consideration holds for the other scenarios studied in the first part of Section 5, so their thorough discussion is omitted.

Let us recall the details of the experiment. We assume independent components of $\theta^0$ and $\nu$ such that

$$\theta^0_i \sim \mathcal{N}(0,1), \quad \nu_j \sim \mathrm{Lap}(0,1),$$
$$i = 1, ..., m, \ j = 1, ..., n.$$

We first implement MCMC to simulate the posterior in (50). Some care is needed to select a suitable proposal density: to this aim, we implement a random walk Metropolis scheme [24, Section 1] with candidates obtained by Gaussian independent increments of standard deviation 0.4 (this ensures an acceptance rate around 30%). A Markov chain of length $10^5$ is then generated to achieve the posterior in sampled form. Quantiles 0.025 and 0.975 are then extracted to obtain 95%-level bounds. Next, Algorithm 1 is used to obtain the bounds. As regards model parameters of equations (2) and (3), we set $B = I_n$, $C = I_m$ and $\mu = 0_{m\times 1}$, and the entries of $A$ are realizations of independent zero-mean Gaussians of unit variance. In this experiment we choose $n = 200$ output samples and a dimension of $\theta^0$ equal to $m = 20$. A sample outcome of the bounds obtained via MCMC and BFB is presented in Figure 5. The bounds look quite similar: BFB still gives a clear picture about the information provided by the outputs. It does not require defining a proposal density, and it provides the result in one tenth of a second. On the other hand, the computational time of MCMC is two orders of magnitude greater, since about $10^3$ samples have to be drawn.