Distributed parametric and nonparametric regression with on-line performance bounds computation

[a] *School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden*

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

Damiano Varagnolo [a] Gianluigi Pillonetto, Luca Schenato [b]

Abstract

In this paper we focus on collaborative multi-agent systems, where agents are distributed over a region of interest and collaborate to achieve a common estimation goal. In particular, we introduce two consensus-based distributed linear estimators. The first one is designed for a Bayesian scenario, where an unknown common finite-dimensional parameter vector has to be reconstructed, while the second one regards the nonparametric reconstruction of an unknown function sampled at different locations by the sensors. Both of the algorithms are characterized in terms of the trade-off between estimation performance, communication, computation and memory complexity. In the finite-dimensional setting, we derive mild sufficient conditions which ensure that distributed estimator performs better than the local optimal ones in terms of estimation error variance. In the nonparametric setting, we introduce an on-line algorithm that allows the agents to simultaneously compute the function estimate with small computational, communication and data storage efforts, as well as to quantify its distance from the centralized estimate given by a Regularization Network, one of the most powerful regularized kernel methods. These results are obtained by deriving bounds on the estimation error that provide insights on how the uncertainty inherent in a sensor network, such as imperfect knowledge on the number of agents and the measurement models used by the sensors, can degrade the performance of the estimation process. Numerical experiments are included to support the theoretical findings.

Key words: distributed learning, regularization, Gaussian processes, parametric estimation, nonparametric estimation, wireless sensor networks, reproducing kernel Hilbert spaces, consensus

1 Introduction

New low-cost technologies and wireless communication are promoting the deployment of networks with a large number of sensors (often called also *nodes*, or *agents*) which can communicate and collaborate to achieve a common objective. These networks, whose popularity and diffusion is increasing, can be employed for a wide range of applications such as remote surveillance of hazardous areas, environmental monitoring, and indoor target tracking [1,2]. Although these networks promise incredible advantages as compared to more traditional technologies, they also pose challenging novel questions from both theoretical and practical perspectives in terms of information compression [3], distributed learning [4], and event detection [5,6], just to name a few.

1.1 Literature review

Over the multitude of different multi-agent systems, we focus on collaborative ad-hoc wireless sensor networks, i.e., networks in which sensors are randomly distributed over a region of interest and collaborate to achieve a common goal [7]. We assume that agents have limited computational and communication capabilities, that there are no central coordinating units or fusion centers, and that each sensor aims at obtaining a shared knowledge close to the one computable through a centralized strategy. Finally we assume that the topology can be dynamic, allowing agents to randomly appear, disappear or move. For this reason we will let the nodes have only a limited topological knowledge. In particular we assume that

⋆ The research leading to these results has received funding from the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement n°257462 HYCON2 Network of excellence and n°223866 FeedNetBack, from Progetto di Ateneo CPDA090135/09 funded by the University of Padova, and by the Italian PRIN Project "New Methods and Algorithms for Identification and Adaptive Control of Technological Systems", from the Knuth and Alice Wallenberg Foundation and the Swedish Research Council.

Email addresses: damiano@kth.se (Damiano Varagnolo), { giapi | schenato }@dei.unipd.it (Gianluigi Pillonetto, Luca Schenato).

they only know the probability density of their physical location. Examples of such networks are wireless sensor networks for forest-monitoring where identical sensors are dropped from an helicopter, or networks of mobile robots exploring an unknown but limited region.

In this paper we focus on the problem of distributed regression [8,9] subject to communication and computational constraints for parametric [10] and nonparametric [11] models. In the context of parametric estimation, several distributed strategies have been proposed. For example, in [12] the authors focus on consensus-based decentralized estimation of deterministic parameter vectors, considering both Maximum Likelihood (ML) and Best Linear Unbiased Estimation (BLUE) schemes, solved through a set of convex minimization subproblems. Distributed convex optimization has also been used in [13] to distributedly compute the Linear Minimum Mean Square-Error (LMMSE) estimate of an unknown signal through the parallelization of coordinate descent steps. Similar techniques have been used in [14], where authors consider three different consensus-based distributed Lasso regression algorithms: the first based on quadratic programming techniques, the second on cyclic coordinate descent steps, and the third on the decomposition of the original cost function into smaller optimization subproblems. Other authors proposed distributed inference schemes based on graphical models, like in [15] or in [16], offering an LMMSE estimator that exploits a particular implementation of the Gauss-Seidel matrix inversion algorithm.

Parametric descriptions of processes naturally arise in scenarios where it is possible to formulate a specific class of finite-dimensional models such as polynomials or radial basis functions. However, there are problems for which this is difficult and nonparametric estimation has been found to be more suitable and effective. In particular, the nonparametric approaches have been proved to be consistent with respect of a large number of models classes, such as the NARX models [17]. Within this framework, the theory of reproducing kernel Hilbert spaces (RKHSs) [18] has been often used for regression purposes [19,20]. This theory has been successfully used also in distributed scenarios: for example, [21] considers the problem of jointly estimating time delays and functions, while [22] proposes a distributed regularized kernel Least Squares (LS) regression problem based on successive orthogonal projections. Similarly, in [23] the authors extend [22] by proposing modifications reducing the communication burden and synchronization assumptions. In [24], a reduced order model approach is proposed, where sensors construct an estimate considering only a subset of the representing functions that would be used to form the optimal solution. Other approaches involve message-passing based schemes in graphical models: in [25] the authors consider a nonparametric distributed regression algorithm that is subject to communication constraints, while [26] considers kernel linear regressors without regularizing terms.

Although the current trend is towards the design of purely distributed algorithms where each agent runs the same algorithm, also hierarchical strategies have been proposed. For example, [27] offers a distributed Bayesian learning scheme where a supervisor node fuses the results of local outputs, [28] proposes an iterative conditional expectation algorithm that distributedly estimates a deterministic function, while [29] uses a pre-defined cyclic learning schemes based on information routing tables.

An other interesting research field is given by mobile sensor networks, where agents exploit their motion capabilities to perform particular tasks. Examples are [30], where the author introduces the so-called Distributed Kriged Kalman Filter, an algorithm used to estimate the distribution of a dynamic Gaussian random field and its gradient. We notice that in [30] sensors estimate their own neighborhood and not to the global scenario. In the same framework, in [31] authors develop a distributed learning and cooperative control algorithm where sensors estimate a static field. The field is modeled as a network of radial basis functions whose number and centers location are known in advance by sensors. Nonparametric schemes are applied also in [32], where the mobile sensor network distributedly estimates a noisily sampled scalar random field through opportune Nearest-Neighbors interpolation schemes, and in [33], where the authors use subsets of measurements to perform Gaussian processes based regression and robots coordination.

1.2 Contribution

In this work, we consider a situation where each sensor collaborates to estimate a global unknown parameter vector or function, so that at the end of the process each node will have the same copy of the estimate. We propose several distributed estimation algorithms, for both the parametric and the nonparametric scenario. All the proposed numerical schemes are characterized in terms of the trade-off between estimation performance and communication, computation, memory complexity. In particular, regarding the estimation performance, two types of quantitative bounds are provided. The first types of bounds are derived in a Bayesian parametric scenario, are related to the estimation error variance and can be computed off-line, i.e., before the measurements become available to the sensors. They tend to be pessimistic but permit to gain insights on how the uncertainty inherent in the network, such as the imperfect knowledge on the number of agents, can degrade the performance of the estimation process. The second types of bounds obtained in this paper are more general under many aspects. The non-parametric strategy adopted here is inspired by the recent studies on approximated regularization methods contained in [34], but it is extended to include a more general model for the measurement process and to be applicable to distributed multi-agent systems. More specifically, rather than a Bayesian setting, we instead con-

sider a scenario where the unknown function is deterministic and belongs to a possibly infinite-dimensional space. In accordance with the modern paradigm of statistical learning theory, see [35], this permits to obtain bounds that are not affected by possible statistical modeling errors on the function to reconstruct. Moreover, each sensor collects noisy measurements at different locations, which are unknown to the other sensors. The sole knowledge shared by the nodes are the stochastic mechanism from which the sampling locations are drawn and the size of the hypotheses space, that can be computed off-line exploiting opportune guidelines. Finally, the bounds are computed on-line, after the measurement process is performed, adding some extra complexity. However, they turn out to be accurate in quantifying the distance between the estimate of the proposed distributed version and the estimate obtainable by a centralized version of a Regularization Network (RN) [20,36], one of the most powerful and used regularized kernel methods.

In particular, our bounds account for the uncertainty in the number of sensors present in the network, the uncertainty relative to the different measurement models used by the agents as well as the infinite-dimensional nature of the hypothesis space. Differently from the other distributed learning approaches present in the literature and listed above, our scheme is able to to provide a certificate of quality of the estimate on a non-asymptotic basis.

It is important to notice that the techniques we propose rely on computation of averages, that can be distributedly computed through consensus algorithms [37,38,39,40]. This is attractive because of their simplicity, their completely asynchronous communication schemes, their robustness to nodes and links failures, their scalability with the network size, and the absence of a coordination unit. In what follows, we will assume that the communication graph is sufficiently connected in order to allow the computation of consensus algorithms [41] and that a sufficient number of consensus steps are performed to guarantee convergence to the true average. In addition, we consider the static scenario, where distributed inference is indeed a topic of recent interest within the control's community, e.g., see [30,31,32].

The paper is structured as follows: Section 2 introduces the parametric distributed estimator, that is characterized in Section 3. Section 4 deals with the nonparametric scenario and proposes a number of estimators, whose tuning and quantification of prediction capabilities are offered in Sections 5 and 6, respectively. The theoretical results are complemented with numerical simulations in Section 7, while in Section 8 some concluding remarks are drawn. In the interest of clarity, all the proofs of the propositions and theorems are gathered in the Appendix.

2 Parametric distributed estimation algorithms

In this section we consider S distinct sensors each of them taking M scalar noisy measurements on the same input locations. We model this scenario in a parametric framework as

$$y_i = Cb + \nu_i, \qquad i = 1, \dots, S \tag{1}$$

where $y_i \in \mathbb{R}^M$ is the measurements vector collected by the i-th sensor, and $b \in \mathbb{R}^E$ is the vector of unknown parameters modeled as a zero-mean Gaussian vector with autocovariance Λ_0, i.e., $b \sim \mathcal{N}(0, \Lambda_0)$. In addition, $\nu_i \in \mathbb{R}^M$ is the noise vector with density $\mathcal{N}\left(0, \sigma^2 I\right)$, independent of b and of ν_j, for $i \neq j$. Finally, $C \in \mathbb{R}^{M \times E}$ is a known matrix identical for all sensors.

The more general scenario where the variances of the measurement noise may be different among sensors, i.e., $\nu_i \sim \mathcal{N}\left(0, \sigma_i^2 I\right)$, was addressed in [42].

2.1 Local Bayesian estimation

Under the assumptions above, the *local* Minimum Mean Square Error (MMSE) estimator of b given y_i is unbiased and given by [10]

$$\begin{aligned} b_\ell^i &:= \mathbb{E}\left[b \,|y_i\right] = \operatorname{cov}\left(b, y_i\right) \left(\operatorname{var}\left(y_i\right)\right)^{-1} y_i \\ &= \Lambda_0 C^T \left(C \Lambda_0 C^T + \sigma^2 I\right)^{-1} y_i = K_\ell \, y_i \\ &= \left(\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)^{-1} \frac{C^T y_i}{\sigma^2} = H_\ell \, i_i \end{aligned} \tag{2}$$

where $i_i := C^T y_i / \sigma^2$, while the autocovariance of the local estimation error is

$$\Lambda_\ell^i = \Lambda_\ell := \operatorname{var}\left(b - b_\ell^i\right) = \left(\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)^{-1} \tag{3}$$

and is independent of the measurements y_i. In the distributed estimation framework we assume that each sensor wants to exchange information in order to refine the knowledge about b. The performance metrics will be in terms of the estimation error variance.

2.2 Centralized Bayesian estimation

If $S \geq 2$ and all measurements $\{y_i\}_{i=1}^S$ are collected by a central unit, the MMSE estimate of b given $\{y_i\}_{i=1}^S$ can be computed as

$$b_c := \operatorname{cov}\left(b, \begin{bmatrix} y_1 \\ \vdots \\ y_S \end{bmatrix}\right) \operatorname{var}\left(\begin{bmatrix} y_1 \\ \vdots \\ y_S \end{bmatrix}\right)^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_S \end{bmatrix} \tag{4}$$

where

$$\text{var}\left(\begin{bmatrix} y_1 \\ \vdots \\ y_S \end{bmatrix}\right) = \begin{bmatrix} V\left(\sigma^2\right) & \dots & V\left(0\right) \\ \vdots & & \vdots \\ V\left(0\right) & \dots & V\left(\sigma^2\right) \end{bmatrix} \tag{5}$$

with

$$V(\theta) := C\Lambda_0 C^T + \theta I \, . \tag{6}$$

Note that $V\left(\sigma^2\right) = \text{var}(y_i)$, i.e., $V\left(\sigma^2\right)$ corresponds to the variance of a generic measurements vector y_i. Using the matrix inversion lemma and simple algebraic manipulations, (4) can be rewritten in two equivalent forms, i.e., as

$$b_c = \Lambda_0 C^T \left(C\Lambda_0 C^T + \frac{\sigma^2}{S} I\right)^{-1} \left(\frac{1}{S}\sum_{i=1}^{S} y_i\right) = K_c \bar{y} \tag{7}$$

where $\bar{y} := 1/S \sum_{i=1}^{S} y_i$ and $K_c \in \mathbb{R}^{E\times M}$ or as

$$b_c = \left(\frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)^{-1} \left(\frac{1}{S}\sum_{i=1}^{S} \frac{C^T y_i}{\sigma^2}\right) = H_c \bar{i} \tag{8}$$

where $\bar{i} := 1/S \sum_{i=1}^{S} C^T y_i/\sigma^2$ and $H_c \in \mathbb{R}^{E\times E}$. To compute b_c through (7), sensors need to reach an average consensus on their sets of measurements y_i, which are M-dimensional vectors, while to compute b_c through (8) they need to reach an average consensus on the transformed measurement vectors [1] $i_i = C^T y_i/\sigma^2$ which are E-dimensional vectors.

In order to be consistent with the nonparametric part of the paper, we will consider only (8). Obviously, the variance of the estimation error is the same for both the forms, and is given by

$$\Lambda_c := \text{var}\left(b_c - b\right) = \frac{1}{S}\left(\frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)^{-1} \, . \tag{9}$$

2.3 Distributed Bayesian estimation

To implement the optimal estimation strategy given in (8), sensors need both to reach consensus on their $C^T y_i/\sigma^2$ and also properly weight the contribution of the prior Λ_0. This implies that all the sensors must have perfect knowledge on S, the actual number of agents participating to the consensus process. Since sometimes this request cannot be satisfied, it is interesting to characterize the performance of the approximated distributed estimation strategy

$$b_d\left(S_g\right) := \left(\frac{1}{S_g}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)^{-1} \left(\frac{1}{S}\sum_{i=1}^{S} \frac{C^T y_i}{\sigma^2}\right) = H_d \, \bar{i} \tag{10}$$

where S_g is an estimate of the number of sensors in the network. To simplify the notation, in the following we denote $b_d\left(S_g\right)$ as b_d unless differently stated. Simple algebraic manipulations lead to the computation of the corresponding estimation error covariance

$$\Lambda_d(S_g) := \text{var}\left(b_d - b\right) = H_d \left(\frac{1}{S_g^2}\Lambda_0^{-1} + \frac{1}{S}\frac{C^T C}{\sigma^2}\right) H_d \, . \tag{11}$$

Notice that for $S_g = 1$, then $b_d(1) = 1/S \sum_{i=1}^{S} b_\ell^i$, i.e., the average the local estimators, while for $S_g = +\infty$ then $b_d\left(\infty\right) = \left(S\, C^T C\right)^{-1} \left(\sum_{i=1}^{S} C^T y_i\right)$, i.e., the least squares solution which discards the prior information on b, and finally for $S_g = S$ then $b_d(S) = b_c$, i.e., the centralized solution.

3 Characterization of the parametric distributed estimation algorithm

In this section, we determine conditions on the parameter S_g that guarantee $\Lambda_d(S_g) \leq \Lambda_\ell$, hence ensuring that a distributed strategy sharing information among nodes is better than one using only local information. Moreover, we determine the accuracy of the distributed strategy (10), as compared to the centralized solution (8). These two scenarios are addressed separately.

3.1 Distributed versus local estimation

It is possible to derive the following result.

Theorem 1 *Let d_{min} be the smallest eigenvalue of $C\Lambda_0 C^T$. If one of the two following conditions is satisfied:*

$$(a)\ \frac{(S-1)d_{min}}{\sigma^2} > 1 \qquad (b)\ S_g \in [1, 2S-1] \tag{12}$$

then $\Lambda_d \leq \Lambda_\ell$.

The sufficient condition (12)-(a) implies that the distributed estimator is always better than the local one for all $S_g \in [1, +\infty)$. In particular, the ratio $(S-1)d_{\min}/\sigma^2$ can be interpreted as the smallest signal-to-noise ratio (SNR) among all possible output directions, therefore the inequality states that if the SNR is sufficiently large, then the distributed estimator always provides a better performance than the local one. The second sufficient condition (12)-(b) of the previous Theorem can be considered *universal* since it holds for every prior Λ_0, number of measurements M, number of parameters E, measurement noise variance σ^2, and matrix C. It assures

[1] The vector i_i is also known as the *information vector* associated to the measurement y_i in the context of parametric estimation [43].

that there exists a large set of potential guesses of number of sensors S_g for which the distributed estimator b_d is performing better than the local one b_ℓ, for all possible SNRs. In particular, this theorem confirms the intuition that the average of the local estimators, i.e., $S_g = 1$, always produces a better estimate. Moreover, even rough estimates of S are likely to improve the estimation performance as compared to the local estimator.

Although the conditions in Theorem 1 are only sufficient, they are nonetheless tight, in the sense that there are scenarios for which, if they are not satisfied, then $\Lambda_d > \Lambda_\ell$. This is in fact the case of scalar systems, i.e., $b \in \mathbb{R}$, as shown in Section 3.3.

3.2 Distributed versus centralized estimation

Although we always have $\Lambda_c \leq \Lambda_d$, it is relevant to study the influence of the parameter S_g on the distance between the centralized estimator b_c and the distributed estimator b_d. If prior bounds about the unknown parameter S are available, i.e., $S \in [S_{\min},\ S_{\max}]$, then it is possible to prove the following theorem:

Theorem 2 *Under the assumption that* $S \in [S_{min}, S_{max}]$ *then*

$$\frac{\|b_d - b_c\|_2}{\|b_d\|_2} \leq \frac{S_{max}}{S_{min}} - 1 \tag{13}$$

for all $S_g \in [S_{min}, S_{max}]$.

Although the bound provided in the theorem could be ameliorated if additional knowledge about Λ_0 and C were available, it is tight as shown in the example below.

3.3 Numerical examples

We consider the particular scalar case $E = 1$, $M = 1$, $\Lambda_0 = 1$, $C = 1$ and $S = 100$, which implies $d_{\min} = 1$, and analyze the performance of the distributed estimator as a function of the the measurement noise level σ^2 and the guess S_g. In this special scenario, it is possible to show that the bounds of Theorem 1 and 2 are indeed tight. In fact, according to condition (12)-(a), Theorem 1 is satisfied for $\sigma^2 < \sigma_c^2 := 99$. However, it is easy to verify that if $\sigma^2 = \sigma_c^2$ and $S_g > 2 \cdot 10^4$ then $\Lambda_d(S_g) > \Lambda_\ell$. Similarly, according to condition (12)-(b), $\Lambda_d(S_g) \leq \Lambda_\ell$ for $S_g \leq 2S - 1$, however it can be checked that if $S_g \geq 2S$ and $\sigma^2 > 4 \cdot 10^4$ then $\Lambda_d(S_g) > \Lambda_\ell$. Figure 1 graphically displays the distributed estimator error performance Λ_d as a function of S_g for different values of the measurement noise variance σ, as well the local estimator performance $\Lambda_\ell = \frac{\sigma^2}{\sigma^2+1} \approx 1$.

Under the same scalar scenario, Figure 2 compares the true relative error

$$e_d = \frac{\|b_d - b_c\|_2}{\|b_d\|_2} = \left| \frac{(S - S_g)\,\sigma^2}{(S + \sigma^2)\,S_g} \right| \tag{14}$$

with bound (13) provided in Theorem 2, for different

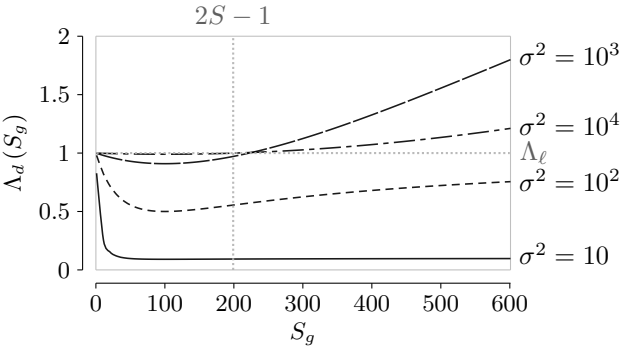

Figure 1. Estimation error variance Λ_d of the distributed estimator b_d defined in (11) as a function of S_g for different values of σ^2, for the particular case $E = 1$, $M = 1$, $\Lambda_0 = 1$, $C = 1$ and $S = 100$. The dotted gray line approximatively indicates the estimation error variance Λ_ℓ of the local estimators.

values of $S_{\max}/S_{\min}$, S_g, S and σ^2. Although this bound is often pessimistic, it is indeed tight, in fact for $S = S_{\max}$ and $S_g = S_{\min}$ then $\lim_{\sigma^2 \to +\infty} e_d = \frac{S_{\max}}{S_{\min}} - 1$.

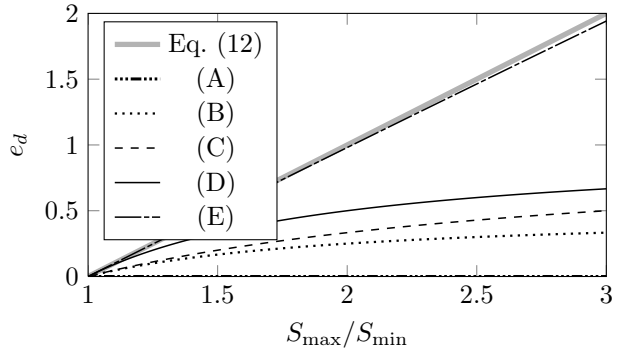

Figure 2. Dependency of the relative error $e_d = \frac{\|b_d - b_c\|_2}{\|b_d\|_2}$ on $S_{\max}/S_{\min}$ and σ^2 for various choices of S_g, for the scenario $E = 1$, $M = 1$, $\Lambda_0 = 1$, $C = 1$. (A): $S_g = S$. (B): $S = S_{\min}$, $S_g = S_{\max}$, $\sigma^2 = 10^2$. (C): $S = S_{\max}$, $S_g = S_{\min}$, $\sigma^2 = 10^2$. (D): $S = S_{\min}$, $S_g = S_{\max}$, $\sigma^2 = +\infty$. (E): $S = S_{\max}$, $S_g = S_{\min}$, $\sigma^2 = 10^4$. "bound": bound (13).

A more realistic scenario would also include uncertainty on the mean and variance of the measurement process. A possible generalization is, for example, to consider a probabilistic model for the mean and variance based on some hyper-parameters, and then find performance bounds on these hyper-parameters. However, these derivations are not straightforward, and we rather propose to address the need of a more realistic measurement processes by considering a non-parametric scenario.

4 Nonparametric distributed function estimation

4.1 Centralized scenario

Let $f_\mu : \mathcal{X} \to \mathbb{R}$ denote an unknown deterministic function defined on the compact $\mathcal{X} \subset \mathbb{R}^d$. Assume there are

S sensors, each collecting a single noisy measurement [2] y_i where

$$y_i = f_\mu(x_i) + \nu_i, \quad i = 1, \ldots, S \tag{15}$$

with ν_i white noise and i the sensor index. We assume that each input location x_i is known only to the i-th sensor and that it is independently drawn from a probability measure μ known to all the sensors.

Given the data set $\{x_i, y_i\}_{i=1}^S$, one of the most used approaches to estimate f_μ relies upon the Tikhonov regularization theory [44]. The hypothesis space is typically given by a reproducing kernel Hilbert space (RKHS) defined by a Mercer Kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ [45,46,47]. In particular, let $\mathcal{L}^2(\mu)$ be the set of the Lebesgue square integrable functions under the measure μ, and define the positive integral operator

$$L_{K,\mu}[g](x) := \int_{\mathcal{X}} K(x, x') g(x') d\mu(x') \ . \tag{16}$$

Its eigenvalues λ_e and eigenfunctions ϕ_e can be computed as , e.g., in [17], and satisfy

$$\lambda_e \phi_e(x) = L_{K,\mu}[\phi_e](x) \tag{17}$$

with $\lambda_1 \geq \lambda_2 \geq \ldots \geq 0$. The following result holds.

Theorem 3 ([48]) *Let K be a Mercer kernel on $\mathcal{X} \times \mathcal{X}$, $\lambda_e > 0 \ \ \forall e$ and μ a non-degenerate measure [3]. Then, $\{\phi_e\}_{e=1}^{+\infty}$ is an orthonormal basis in $\mathcal{L}^2(\mu)$ while the associated RKHS is*

$$\mathcal{H}_K := \left\{ g \in \mathcal{L}^2(\mu) \ \ s.t. \ \ g = \sum_{e=1}^{\infty} \alpha_e \phi_e \ \ with \ \{\alpha_e\} \ s.t. \ \sum_{e=1}^{\infty} \frac{\alpha_e^2}{\lambda_e} < +\infty \right\}. \tag{18}$$

Moreover, if $g_1 = \sum_{e=1}^{+\infty} \alpha_e \phi_e$ and $g_2 = \sum_{e=1}^{+\infty} \beta_e \phi_e$, their inner product is $\langle g_1, g_2 \rangle_{\mathcal{H}_K} = \sum_{e=1}^{+\infty} \frac{\alpha_e \cdot \beta_e}{\lambda_e}$.

Notice that, if $g = \sum_{e=1}^{+\infty} \alpha_e \phi_e \in \mathcal{H}_K$ and $\alpha := [\alpha_1, \alpha_2, \ldots]^T$, orthogonality of eigenfunctions in $\mathcal{L}^2(\mu)$ implies that

$$\|g\|^2_{\mathcal{L}^2(\mu)} = \sum_{i=1}^{+\infty} \sum_{j=1}^{+\infty} \alpha_i \alpha_j \int_{\mathcal{X}} \phi_i(x) \phi_j(x) d\mu(x) = \|\alpha\|_2^2 \ . \tag{19}$$

[2] The assumption of a single measurement per sensor has been posed only for brevity and is not restrictive for our purposes.

[3] A Borel measure μ is said to be non-degenerate w.r.t. the Lebesgue measure $\mathcal{L}^2$ if $\mathcal{L}^2(A) > 0 \Rightarrow \mu(A) > 0$ for every A in the Borel σ-algebra.

In the following, we will use the shorthands $\|\cdot\|_\mu$ for $\|\cdot\|_{\mathcal{L}^2(\mu)}$ and $\|\cdot\|_K$ for $\|\cdot\|_{\mathcal{H}_K}$.

According to the regularization theory, a common choice for the cost function is

$$Q(f) := \sum_{i=1}^{S} (y_i - f(x_i))^2 + \gamma \|f\|_K^2 \tag{20}$$

and the estimate of the unknown function is

$$f_c := \arg \min_{f \in \mathcal{H}_K} Q(f) \ . \tag{21}$$

In (20), γ is the so called *regularization parameter* that trades off empirical evidence and smoothness information on f_μ. It is well known that f_c admits the structure of a Regularization Network, see [49], being the sum of S basis functions with expansion coefficients obtainable by inverting a system of linear equations. More precisely, one has

$$f_c = \sum_{i=1}^{S} c_i K(x_i, \cdot) \ \ , \ \ \begin{bmatrix} c_1 \\ \vdots \\ c_S \end{bmatrix} = (\mathbf{K} + \gamma I)^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_S \end{bmatrix} \tag{22}$$

where

$$\mathbf{K} := \begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_S) \\ \vdots & & \vdots \\ K(x_S, x_1) & \cdots & K(x_S, x_S) \end{bmatrix} . \tag{23}$$

Remark 4 *The estimate f_c in (21) admits also a Bayesian interpretation. In fact, if f_μ is modeled as the realization of a zero-mean Gaussian random field with covariance K, the noise ν_i is Gaussian, independent of the unknown function and with variance σ^2, setting $\gamma = \sigma^2$ one has*

$$f_c(x) = \mathbb{E}\left[f_\mu(x) \left| \{x_i, y_i\}_{i=1}^S \right. \right] \ . \tag{24}$$

Hence, under such Bayesian perspective, the problem of reconstructing f_μ is a generalization of that discussed in Section 2. The increased complexity derives from the fact that the problem is now infinite-dimensional and each sensor has not a complete knowledge of the measurement model of the other agents, since the input location x_i is known only to the i-th sensor.

Notice that the computation of f_c requires $O\left(S^3\right)$ operations and the processing unit has to store all the x_i's. This can be impractical in a distributed estimation scenario, where agents may have both limited computational and communication resources. To overcome these

problems, in Section 4.2 we will derive an alternative distributed estimation strategy by restricting the hypothesis space to a closed subspace $\breve{\mathcal{H}}_K \subset \mathcal{H}_K$. The following proposition shows that the resulting estimator has favorable theoretical properties. In particular, as the number of sensors S goes to $+\infty$, it returns the best possible approximation of f_μ in $\breve{\mathcal{H}}_K$. The result is obtained along the same lines used in [50,51] to characterize the estimator (21).

Proposition 5 *Let $0 < \delta < 1$ and define the closed subspace $\breve{\mathcal{H}}_K := \overline{span_{e\in\mathcal{I}}\{\phi_e\}}$, where the overline denotes the closure in $\mathcal{H}_K$, and where $\mathcal{I} \subset \mathbb{N}_+$. Define* [4]

$$\widehat{f}_S := \arg\min_{f\in\breve{\mathcal{H}}_K} \sum_{i=1}^{S} \frac{(y_i - f(x_i))^2}{S} + \gamma \|f\|_K^2 \ . \quad (25)$$

Then, assuming $|y_i| \leq \overline{Y}$ a.s. [5]*, if $S \to +\infty$ then $\widehat{f}_S$ converges in probability to the projection of f_μ onto $\breve{\mathcal{H}}_K$, denoted by $f_\mu^{\breve{\mathcal{H}}_K}$. In particular, let*

$$\gamma = \frac{8\overline{K}^2 \log\left(\frac{4}{\delta}\right)}{\sqrt{S}} \ , \quad \overline{K} := \sup_{x_1,x_2\in\mathcal{X}} \sqrt{K(x_1,x_2)} \ , \quad (26)$$

$$\overline{D} := \frac{\sqrt{2\log\left(\frac{4}{\delta}\right)}}{S^{\frac{1}{4}}} \left(3\overline{Y} + 2\overline{K} \left\| f_\mu^{\breve{\mathcal{H}}_K} \right\|_K\right) + \left\| f_\mu^{\breve{\mathcal{H}}_K^\perp} \right\|_\mu$$

where $f_\mu^{\breve{\mathcal{H}}_K^\perp}$ is the projection of f_μ onto the orthogonal of $\breve{\mathcal{H}}_K$ in $\mathcal{H}_K$. Then

$$\mathbb{P}\left[\left\| \widehat{f}_S - f_\mu \right\|_\mu \leq \overline{D} \right] \geq 1 - \delta \ . \quad (27)$$

Remark 6 *Despite of the possible stochastic interpretation of the problem recalled in Remark 4, in all this section f_μ always represents an unknown but deterministic function. It is worth stressing that in this scenario the bounds that one obtains, regarding the performance of the proposed estimator, are more robust since are not affected by errors in the statistical modeling of f_μ. This is in accordance with the modern statistical learning theory as described, e.g., in [35]. An example of this has been already provided in Proposition 5 where the validity of (27) just requires f_μ to belong to an infinite-dimensional space that may contain a very wide class of functions. For instance, popular choices for $\mathcal{H}_K$ are Sobolev spaces or spaces induced by the Gaussian kernel which are all known to be dense in the space of continuous functions, e.g., see [52].*

We now consider a particular finite-dimensional subspace $\breve{\mathcal{H}}_K$, denoted by $\mathcal{H}_K^E$, that is generated by the first E eigenfunctions ϕ_e, i.e.,

$$\mathcal{H}_K^E := \left\{ g \in \mathcal{L}^2(\mu) \ \text{ s.t. } \ g = \sum_{e=1}^{E} \alpha_e \phi_e \ \text{ with } \ [\alpha_1, \ldots, \alpha_E]^T \in \mathbb{R}^E \right\} . \quad (28)$$

The particular choice for $\mathcal{H}_K^E$ is motivated by the presence of the penalty term $\|\cdot\|_K^2$ used to obtain the function estimate. It can also be justified using the Bayesian framework described in remark 4 under which, before seeing the data, $\mathcal{H}_K^E$ represents the subspace that captures the biggest part of the signal variance among all the subspaces of dimension E. This is in accordance with the Rayleigh's principle [53,54] which underlies Principal Component Analysis.

Using $\mathcal{H}_K^E$ as hypothesis space, the estimate of f_μ is given by [6]

$$f_r := \arg\min_{f\in\mathcal{H}_K^E} Q(f) \ . \quad (29)$$

As it will be clear in the sequel, it is now convenient to reformulate the estimates f_c and f_r through the map $T : \mathcal{H}_K \to \mathbb{R}^\infty$ that is induced by definition (18) and associates to a function $f = \sum_{e=1}^{+\infty} a_e \phi_e$ the sequence $[a_1, a_2 \ldots]$, i.e., $T[f] = [a_1, a_2, \ldots]^T$. With a little abuse of notation, we also equip $\mathbb{R}^\infty$ with the norm $\|a\|_K^2 := \sum_{e=1}^{+\infty} \frac{a_e^2}{\lambda_e}$ so as to make T an isometric mapping. In what follows, if A is a matrix with an infinite number of columns and rows, and w is a column vector with an infinite number of rows, then Aw is the vector with i-th element equal to $\sum_{j=1}^{\infty} [A]_{ij} w_j$. In addition, A^{-1} denotes the inverse of the operator induced by A, i.e., we use notation of ordinary algebra to handle infinite-dimensional objects.

Exploiting $T[\cdot]$, the measurement model (15) can thus be rewritten as

$$y_i = C_i b + \nu_i \quad i = 1, \ldots, S \quad (30)$$

where $b = T[f_\mu]$ and [7]

$$C_i := [\phi_1(x_i) \ \ \phi_2(x_i) \ \ \ldots] \ . \quad (31)$$

Notice that C_i is a stochastic i.i.d. sequence whose distribution depends on μ. The two following propositions

[4] We notice that (25) is identical to (20) up to a factor S dividing the regularization parameter γ. The choice for this notation has been driven by a desire of consistency with the notation used in [50,51].

[5] This is a standard assumption in the modern statistical learning literature. It is easy to relax it to handle Gaussian noises scenarios, but this would excessively complicate the proofs in a way that is beyond the scope of this paper.

[6] We use the subscript r to recall that f_r lies in a reduced hypothesis space.

[7] We recall that the λ_e's and ϕ_e's can be computed as in [17].

can be obtained as immediate consequences of the results in [34], thus proofs are omitted.

Proposition 7 *Let*

$$Q(b) := \sum_{i=1}^{S} (y_i - C_i b)^2 + \gamma \|b\|_K^2 \ . \tag{32}$$

Then

$$\begin{aligned} b_c &:= \arg\min_b Q(b) \\ &= \left(\operatorname{diag}\left(\frac{\gamma}{\lambda_e}\right) + \sum_{i=1}^{S} C_i^T C_i \right)^{-1} \left(\sum_{i=1}^{S} C_i^T y_i \right) \end{aligned} \tag{33}$$

with $\operatorname{diag}(\alpha_e)$ *to indicate the matrix with diagonal elements given by* $\alpha_1, \alpha_2, \ldots$. *Furthermore,* $T[f_c] = b_c$ *where* f_c *is defined in* (21).

Since (33) involves infinite dimensional vectors, approximated solutions in $\mathcal{H}_K^E$ are now searched. To this aim, defining

$$C_i^E = C^E(x_i) := [\phi_1(x_i), \cdots, \phi_E(x_i), 0, 0, \ldots] \tag{34}$$

the following proposition is obtained.

Proposition 8 *Let*

$$Q^E(b) := \sum_{i=1}^{S} \left(y_i - C_i^E b\right)^2 + \gamma \|b\|_K^2 \ . \tag{35}$$

Then

$$\begin{aligned} b_r &:= \arg\min_b Q^E(b) \\ &= \left(\operatorname{diag}\left(\frac{\gamma}{\lambda_e}\right) + \sum_{i=1}^{S} \left(C_i^E\right)^T C_i^E \right)^{-1} \left(\sum_{i=1}^{S} \left(C_i^E\right)^T y_i \right) \end{aligned} \tag{36}$$

and $T[f_r] = b_r$ *where* f_r *is defined in* (29).

Notice that even if $\left(C_i^E\right)^T C_i^E$ is an infinite dimensional matrix, only its $E \times E$ upper-left block can contain non-zero elements. In the same way, every infinite dimensional vector $\left(C_i^E\right)^T y_i$ can have non-zero elements only in its first E components. This implies that also b_r can have non-zero elements only in its first E components.

4.2 Distributed scenario

The steps developed in the previous section are reminiscent of the work [34]. In this section, these steps are used as a starting point for a different analysis whose aim is to understand how (33) can be distributedly computed. We also propose an additional estimator which is simpler to compute in a distributed scenario and we quantify its performance. To this aim, the expression for b_r given in (36) can now be rewritten in a form suited to distributed estimation, i.e.,

$$b_r = \left(\frac{1}{S} \operatorname{diag}\left(\frac{\gamma}{\lambda_e}\right) + \frac{1}{S} \sum_{i=1}^{S} \left(C_i^E\right)^T C_i^E \right)^{-1} \left(\frac{1}{S} \sum_{i=1}^{S} \left(C_i^E\right)^T y_i \right) \tag{37}$$

This shows that b_r can be distributedly computed through two parallel average consensus algorithms, one on $\left(C_i^E\right)^T C_i^E$ and one on $\left(C_i^E\right)^T y_i$). However, practical implementation of (37) may still be problematic. In fact, the agents must know the exact number of measurements/sensors S. In addition, the amount of information that needs to be transmitted could be too elevated, since it scales with the square of E. For these reasons, it is useful to define another approximation of b_c (and b_r) as follows

$$b_d := \left(\frac{1}{S_g} \operatorname{diag}\left(\frac{\gamma}{\lambda_e}\right) + I \right)^{-1} \left(\frac{1}{S} \sum_{i=1}^{S} \left(C_i^E\right)^T y_i \right) \ . \tag{38}$$

Notice that b_d is an approximation of b_r since

(1) parameter S weighting the regularization term $\operatorname{diag}(\gamma/\lambda_e)$ is replaced with a guess (or estimate) S_g;

(2) $\frac{1}{S} \sum_{i=1}^{S} \left(C_i^E\right)^T C_i^E$ is replaced with $\mathbb{E}_\mu \left[\left(C_i^E\right)^T C_i^E \right]$. In fact, $\mathbb{E}_\mu \left[\left(C_i^E\right)^T C_i^E \right] = I$ because one has

$$\left[\frac{1}{S} \sum_{i=1}^{S} \left(C_i^E\right)^T C_i^E \right]_{mn} = \frac{1}{S} \sum_{i=1}^{S} \phi_m(x_i) \phi_n(x_i) \tag{39}$$

and, in addition,

$$\begin{aligned} &\frac{1}{S} \sum_{i=1}^{S} \phi_m(x_i) \phi_n(x_i) \xrightarrow{S \to +\infty} \\ &\to \int_{\mathcal{X}} \phi_i(x) \phi_j(x) \, d\mu(x) = \delta_{ij} \end{aligned} \tag{40}$$

due to the orthogonality of eigenfunctions in $\mathcal{L}^2(\mu)$ and the fact that the x_i's are i.i.d. and extracted from μ.

4.3 Complexity analysis

It is important to evaluate the tradeoffs in terms of computational, communication and memory complexity for the proposed estimators. The centralized estimator b_c given in (33) is equivalent to the estimator given in (22). It requires the collection and storage at each node of all measurements y_i and input locations x_i, which accounts for communication and memory complexity of $O(S)$, since we assume a fixed number of measurements per node. The computational complexity is dominated by the inversion of the matrix $\mathbf{K} + \gamma I$ which

has the size of the number of measurements, therefore it is of order $O(S^3)$. The estimator b_r given in (37) relies on average consensus algorithms to compute the averages $1/S\sum_{i=1}^{S}\left(C_i^E\right)^T C_i^E$ and $1/S\sum_{i=1}^{S}\left(C_i^E\right)^T y_i$. Since typical average consensus algorithms require the storage and exchange of quantities with the same size of the desired averages at each iteration [40], if they are performed for a fixed number of iterations, then the communication and memory complexity are given by $O(S^2)$ which is the size of $1/S\sum_{i=1}^{S}\left(C_i^E\right)^T C_i^E$. The computational complexity is dominated by the inversion of a matrix of size E, and it is therefore of order $O(E^2)$. Finally, the estimator b_d given in (38) requires only the computation of the average $1/S\sum_{i=1}^{S}\left(C_i^E\right)^T y_i$ and the inversion and multiplication of a diagonal matrix with a vector of size E, therefore its communication, memory and computational complexity is of order $O(E)$. These considerations are summarized in Table 1.

estimator	*comput. cost*	*commun. cost*	*memory cost*
b_c (Eq. (33))	$O\left(S^3\right)$	$O\left(S\right)$	$O\left(S\right)$
b_r (Eq. (36))	$O\left(E^3\right)$	$O\left(E^2\right)$	$O\left(E^2\right)$
b_d (Eq. (38))	$O\left(E\right)$	$O\left(E\right)$	$O\left(E\right)$

Table 1
Computational, communication and memory costs associated to the introduced estimators.

5 Selection of the number of eigenfunctions E

Let $\overline{E}$ be the maximum admissible value for E, given by computational complexity and transmission capability constraints. Since setting E to $\overline{E}$ could lead to resource wasting, in this section we derive some guidelines for a possibly more parsimonious choice exploiting the a priori information that can be available to the user. We remark that the strategies reported below provide only practical indications on the choice of E before seeing the data. However, the choice of E can be validated using Algorithm 3 developed in Section 6 after seeing the data.

There are mainly two ways for tuning E before seeing the data. The first is based only on the kernel K and selects E based on the cumulative energy of its eigenvalues, as summarized in Algorithm 1.

Algorithm 1 Selection of E (First strategy)

1: Choose a threshold $\varepsilon \in (0,1)$, corresponding to select a pre-defined fraction of the approximation capabilities of the base $\{\phi_i\}_{i=1}^{\infty}$;
2: compute $E = \min \mathcal{E}$ s.t. $\sum_{e=1}^{\mathcal{E}} \lambda_e \geq \varepsilon \sum_{e=1}^{+\infty} \lambda_e$.

Differently, the second strategy tries to include also the prior information about the probability measure μ from which the input locations are drawn. This strategy selects E based on a approximate bound of relative distance between b_c and b_r. Exploiting inequality (A.34) in the proof of Proposition 9 and the equivalence $y_i - C_i b_r = (y_i - C_i b_\mu) + (C_i b_\mu - C_i b_r)$, where $b_\mu = T[f_\mu]$ is the true signal defined in (15), we can write

$$\frac{\|b_c - b_r\|_2}{\|b_r\|_2} \leq \sum_{i=1}^{S} \left\| \mathrm{diag}\left(\frac{\lambda_e}{\gamma}\right) \left(C_i^{\setminus E}\right)^T \right\|_2 \xi_i \qquad (41)$$

$$\text{where} \quad \xi_i := \frac{\|\nu_i\|_2}{\|b_r\|_2} + \frac{\|C_i\left(b_\mu - b_r\right)\|_2}{\|b_r\|_2}.$$

We start assuming that the errors $\|C_i b_\mu - C_i b_r\|_2$ are smaller than κ times the standard deviation of the measurement noise. Therefore, κ regulates the degree of conservativeness of this assumption. A reasonable choice is $\kappa = 3$. We also use the inverse of the SNR as an approximation of both $\frac{\|\nu_i\|_2}{\|b_r\|_2}$ and $\frac{\sigma}{\|b_r\|_2}$. Hence, ξ_i is replaced by $(\kappa+1)\mathrm{SNR}^{-1}$. Finally, for S sufficiently large the quantity

$$S_{\max}\mathbb{E}\left[\left\| \mathrm{diag}\left(\frac{\lambda_e}{\gamma}\right) \left(C_i^{\setminus E}\right)^T \right\|_2\right]$$

overestimates $\sum_{i=1}^{S} \left\| \mathrm{diag}\left(\frac{\lambda_e}{\gamma}\right) \left(C_i^{\setminus E}\right)^T \right\|_2$. The expectation above can be computed using Monte Carlo techniques up to a desired level of accuracy. All the arguments above lead to the following Algorithm 2. Once again it is important to remark that only a rough estimate of E is necessary at this stage and that it can be validated a-posteriori based on the performance analysis provided in Section 6.

Algorithm 2 Selection of E (Second strategy)

1: Assume to have a bound on the SNR, choose a threshold ε for the maximal tolerable error $\frac{\|b_c - b_r\|_2}{\|b_\mu\|_2}$ and choose a degree of conservativeness κ;
2: compute the minimal value of E s.t.

$$(\kappa+1)\mathrm{SNR}^{-1} S_{\max}\mathbb{E}\left[\left\| \mathrm{diag}\left(\frac{\lambda_e}{\gamma}\right) \left(C_i^{\setminus E}\right)^T \right\|_2\right] \leq \varepsilon \qquad (42)$$

where we remark the dependence of the expectation on E.

6 Assessment of the quality of the estimates

Once the choice of E is set, then it is crucial to assess the quality of b_d in terms of its closeness to the optimal centralized estimate b_c. To this regard, the following two results, namely Algorithm 3 and the related Proposition 9, represent the main results of this section. They provide a way to compute, in a distributed fashion, sta-

tistical bounds for the relative errors

$$\|b_d - b_c\|_2 / \|b_d\|_2 \qquad \text{and} \qquad \|b_d - b_r\|_2 / \|b_d\|_2$$

which, in view of (19) and letting $f_d = T^{-1}[b_d]$, coincide respectively with

$$\|f_d - f_c\|_\mu / \|f_d\|_\mu \qquad \text{and} \qquad \|f_d - f_r\|_\mu / \|f_d\|_\mu \ .$$

In the sequel, let

$$C_i^{\setminus E} := [0 \cdots 0 \, \phi_{E+1}(x_i) \ \phi_{E+2}(x_i) \cdots] \ . \tag{43}$$

Furthermore, to compact the notation, let

$$V_r := \left(\frac{1}{S} \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) + \frac{1}{S} \sum_{i=1}^{S} \left(C_i^E \right)^T C_i^E \right)^{-1} \tag{44}$$

$$V_d := \left(\frac{1}{S_g} \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) + I \right)^{-1} \ . \tag{45}$$

Algorithm 3 Distributed estimation and approximation quality evaluation

Off-line work: Sensors are given a level of confidence $1-\delta$, e.g., $\delta = 0.1$, and know $S_{\min}$, $S_{\max}$, S_g, μ, E, as well as the quantity

$$U_S^* := \left(\frac{1}{S_{\min}} - \frac{1}{S_{\max}} \right) \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) \ . \tag{46}$$

In addition, each sensor i stores a particular scenario of the network, i.e., it locally generates $S_{\min}$ independent virtual input locations $x_{i,j}$ by means of density μ

$$x_{i,j} \sim \mu \quad \text{where} \ \ j = 1, \ldots, S_{\min} \tag{47}$$

and then compute the following quantities

$$C_{i,j}^E := [\phi_1(x_{i,j}), \ldots, \phi_E(x_{i,j})] \tag{48}$$

$$V_{r,i}^* := \left(\frac{1}{S_{\max}} \mathrm{diag}\left(\frac{\gamma}{\lambda_e} \right) + \frac{1}{S_{\max}} \sum_{j=1}^{S_{\min}} \left(C_{i,j}^E \right)^T C_{i,j}^E \right)^{-1} \tag{49}$$

$$U_{C,i}^* := \left(I - \frac{1}{S_{\min}} \sum_{j=1}^{S_{\min}} \left(C_{i,j}^E \right)^T C_{i,j}^E \right) \ . \tag{50}$$

▷ continues in the next page

Proposition 9 *Consider Algorithm 3 and the definitions therein. Then, conditioned on z and r_{ave}, it holds that*

$$\mathbb{P}\left(\frac{\|f_r - f_d\|_\mu}{\|f_d\|_\mu} \le \overline{d}_{|dr|}(\delta) \right) \ge 1 - \delta \tag{60}$$

▷ continuation of Algorithm 3

On-line and distributed work:

1: (distributed) sensors distributedly compute, by means of average consensus protocols, the E-dimensional vector $z := \frac{1}{S} \sum_{i=1}^{S} \left(C_i^E \right)^T y_i$

2: (local) each sensor i computes the estimate $b_d = V_d z$

3: (local) each sensor i computes the auxiliary scalars

$$r_i := \frac{\left\| \mathrm{diag}\left(\frac{\lambda_e}{\gamma} \right) \left(C_i^{\setminus E} \right)^T \left(y_i - C_i^E b_d \right) \right\|_2}{\|b_d\|_2} \tag{51}$$

$$s_i := \left\| \mathrm{diag}\left(\frac{\lambda_e}{\gamma} \right) \left(C_i^{\setminus E} \right)^T C_i^E \right\|_2 \tag{52}$$

$$d_{|dr|,i}^* := \frac{\left\| V_{r,i}^* U_S^* b_d \right\|_2}{\|b_d\|_2} + \frac{\left\| V_{r,i}^* U_{C,i}^* b_d \right\|_2}{\|b_d\|_2} \tag{53}$$

4: (distributed) sensors distributedly compute, by means of average consensus protocols, the scalars

$$r_{\text{ave}} := \frac{1}{S} \sum_{i=1}^{S} r_i \qquad s_{\text{ave}} := \frac{1}{S} \sum_{i=1}^{S} s_i \tag{54}$$

$$d_{|dr|,\text{ave}}^* := \frac{1}{S} \sum_{i=1}^{S} d_{|dr|,i}^* \tag{55}$$

$$d_{|dr|,\text{sq}}^* := \frac{1}{S} \sum_{i=1}^{S} \left(d_{|dr|,i}^* \right)^2 \tag{56}$$

5: (local) each sensor i computes

$$d_{|dr|,\text{var}}^* := d_{|dr|,\text{sq}}^* - \left(d_{|dr|,\text{ave}}^* \right)^2 \tag{57}$$

$$\overline{d}_{|dr|}(\delta) := d_{|dr|,\text{ave}}^* + \sqrt{\left(\frac{1}{\delta} - 1 \right) d_{|dr|,\text{var}}^*} \tag{58}$$

$$\overline{d}_{|dc|}(\delta) := S_{\max} r_{\text{ave}} + \overline{d}_{|dr|}(\delta) \left(1 + S_{\max} s_{\text{ave}} \right) \ . \tag{59}$$

$$\mathbb{P}\left(\frac{\|f_c - f_d\|_\mu}{\|f_d\|_\mu} \le \overline{d}_{|dc|}(\delta) \right) \ge 1 - \delta \ . \tag{61}$$

Remark 10 *The results of Proposition 9 are intrinsically different from the ones of Proposition 5. While the latter refers to the consistency of estimators in generic closed subspaces, i.e., to the distance between the estimand and the estimates, the former refers to the degree of approximation committed by discarding the eigenfunctions $\phi_{E+1}, \ldots$. We also notice that these are a posteriori bounds, being computed after the measurement process. A priori bounds, that could be computed similarly, generally return overly pessimistic and thus useless results.*

The proof of Proposition 9 is reported in Appendix and clarifies the quantities that influence the approximation error. In particular, here we just recall that defining

$$U_C := I - \frac{1}{S}\sum_{i=1}^{S}\left(C_i^E\right)^T C_i^E \,, \tag{62}$$

$$U_S := \left(\frac{1}{S_g} - \frac{1}{S}\right)\mathrm{diag}\left(\frac{\gamma}{\lambda_e}\right) \tag{63}$$

$$\text{it follows} \quad \left\|f_d - f_r\right\|_\mu \leq \left\|V_r U_S b_d\right\|_2 + \left\|V_r U_C b_d\right\|_2 \,, \tag{64}$$

i.e., the error between f_d and f_r decomposes into two distinct parts, one involving U_S, proportional to the uncertainty on the number of sensors, and one involving U_C [8], related to the uncertainty on the actual input locations x_i.

Moreover, for what regards the distance of f_d from the optimal estimate f_c, it holds that

$$\begin{aligned}\left\|f_d - f_c\right\|_\mu \leq \left(s_{\text{ave}} S_{\max} + 1\right)\left\|b_d - b_r\right\|_2 + \\ + \textstyle\sum_{i=1}^{S}\left\|\mathrm{diag}\left(\frac{\lambda_e}{\gamma}\right)\left(C_i^{\setminus E}\right)^T\left(y_i - C_i^E b_d\right)\right\|_2 \,,\end{aligned} \tag{65}$$

i.e., the error between f_d and f_c contains the two components described in (64) (scaled by a multiplicative factor always greater than one) plus a term dependent on the sum of the residuals multiplied by a quantity that accounts for the approximation error deriving from replacing $\mathcal{H}_K$ with $\mathcal{H}_K^E$. Notice that the various r_i and s_i defined in (51) and (52) are infinite dimensional quantities but can be locally computed up to the desired accuracy using a finite number of operations.

Summarizing, the stochastic interpretation of Remark 4 assures the approximation capabilities of (21) and (33) to be the ones of centralized Gaussian processes kernel regression strategies. Since distributed implementations are not feasible, we obtain the novel estimators (36) and (38) relying on opportune approximations. Thanks to Proposition 9 and Algorithm 3, agents can then evaluate on-line these performances losses (feature not supported by other techniques, e.g., [33]).

For what regards possible extensions of the algorithm described above, first notice that sensors aiming for more precision on the bounds may locally generate several instances of $d^*_{|dr|,i}$ and then estimate the bounds with the desired level of accuracy. In addition, we also remark that the assumptions on the independence of the various x_i's can be relaxed. In particular, Algorithm 3 and Proposition 9 can be easily extended to handle the case of sensors moving according to an ergodic Markov chain (e.g., generated by the Metropolis-Hastings algorithm [56]) having as invariant measure the desired distribution μ. Finally, it is worth stressing that the entire numerical procedure proposed in this work can also be easily modified to permit the optimization of the regularization parameters present in (33), (36) and (38). For example, the value of γ can be discretized into a finite number of values common to all the sensors and then the "optimal" value can be chosen as the one that minimizes the distance bounds presented above. We eventually remark that the kernel K is given before the measurement process. Learning the kernel from the measurements has received a noticeable attention in centralized scenarios, see, e.g., [57]. It is nonetheless still an open problem in distributed ones.

[8] For an interesting bound on the norm of matrices of the type U_C, as a function of the number of sensors S and of the dimension E, the reader is also referred to Lemma 1 in [55].

7 Numerical examples

Let $f_\mu : [0,1]\times[0,1] \to \mathbb{R}$ be defined by

$$f_\mu\left(x_1, x_2\right) = \beta \sum_{n=1}^{100} \alpha_n \sin\left(\left[\omega_n' \; \omega_n''\right]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) \tag{66}$$

with $\alpha_n \sim \mathcal{N}\left(0, 0.01\right)$ i.i.d., $\omega_n', \omega_n'' \sim \mathcal{U}\left[0, 15\right]$ i.i.d., $\mu \sim \mathcal{U}\left[0,1\right]\times\mathcal{U}\left[0,1\right]$. We will consider a Monte Carlo scenario where at any run the function f_μ is defined by independent realizations of the parameters $\alpha_n, \omega_n', \omega_n''$ and has to be reconstructed from direct measurements corrupted by noise with standard deviation $\sigma = 2$. The parameter β in (66) is adjusted at each run in order to ensure that

$$\text{SNR} := \frac{\int_{\mathcal{X}}\left(f_\mu - \int_{\mathcal{X}} f_\mu d\mu\right)^2 d\mu}{\sigma^2} = 2 \,.$$

To reconstruct f_μ, we use the Gaussian kernel

$$K\left(x_1, x_2; x_1', x_2'\right) = \exp\left(-\frac{\left(x_1 - x_1'\right)^2 + \left(x_2 - x_2'\right)^2}{0.02}\right) \tag{67}$$

where $\gamma = 1$ defines the estimators (33) and (38). Note that the function to be estimated f_μ does not belong to the space generated by this kernel.

The number of eigenfunctions E is obtained by considering the algorithms proposed in Section 5. Figure 3 shows the values of E returned by Algorithms 1 and 2, applied to the following experiment of Figure 5, and fed with various values for the threshold ε. We notice that the exponential decays are inherited by the exponential decay of eigenvalues λ_e associated to the Gaussian kernel. The number E that will be used in the simulations below is selected based on Algorithm 2 by setting $\varepsilon = 0.05$ (indicated in figure with a gray dashed line), which returns $E \approx 100$. We recall that the threshold ε has different

meanings for the two algorithms: in Algorithm 1 it indicates the fraction of the approximation capabilities of the base $\{\phi_i\}_{i=1}^{\infty}$; in Algorithm 2 it indicates a bound on the relative distance $\frac{\|\nu_i\|_2}{\|b_r\|_2}$.

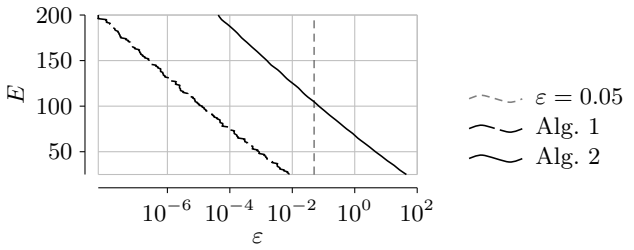

Figure 3. Values of E returned by Algorithms 1 and 2 fed with various choices of the threshold ε and applied to the experiment of Figure 5 with $S_{\max} = 2100$, SNR $= 2$ and $\kappa = 2$. The vertical gray dashed line indicates $\varepsilon = 0.05$.

As for the algorithm proposed in Section 6, we consider 100 runs, i.e., 100 independent realizations [9] of f_μ each sampled by $S = 2000$ sensors (measurements) and estimated using $E = 100$ eigenfunctions. At each run we apply Algorithm 3 to compute the quantities $d^*_{|dr|,\text{ave}}$ and $d^*_{|dr|,\text{var}}$, defined in (55) and (57). With $d^*_{|dr|,\text{ave}}$ and $d^*_{|dr|,\text{var}}$ we then compute two versions of bound (59): the means $\overline{d}_{|dc|}(1)$ and the means plus three standard deviations $\overline{d}_{|dc|}(0.1)$.

Figure 4 plots the true normalized distances between the centralized estimates f_c and the distributed ones f_d versus the relative bounds computed through Algorithm 3, i.e., the points $\left(\frac{\|f_d - f_c\|_\mu}{\|f_d\|_\mu}, \overline{d}_{|dc|}(\delta)\right)$ for different values of the parameter δ and for different uncertainties on the number of sensors, i.e., $S_{\max}, S_{\min}$. In the left panel, the uncertainty on S is smaller ($S_{\min} = 1900, S_{\max} = 2100$) than the one on the right panel ($S_{\min} = 1500, S_{\max} = 2500$).

We remark that the choice $\delta = 1$ corresponds to compute bound (59) exploiting just the average of the local bounds (53). To set $\delta = 0.1$ corresponds instead to a more conservative choice, where (59) is computed from the average of the (53)s plus 3 times their (empirical) standard deviation.

In both panels, the points corresponding to $\delta = 1$ are near the bisector of the first quadrant (black dashed line). This means that bound (59) is in this case significative. Its conservativeness is graphically given by the vertical distance of the points with the bisector. If the bound is computed exploiting also 3 standard deviations of the (53)s ($\delta = 0.1$) then the conservativeness of the

[9] In the interest of clarity, instead of writing $f_{\mu,j}$ with $j = 1, \ldots, 100$, we will used the simplified notation f_μ, unless differently stated. The same applies for the other quantities considered in this section.

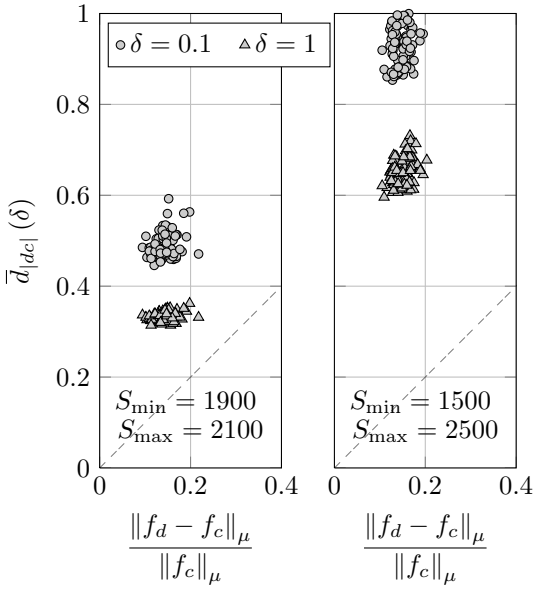

Figure 4. Scatter plot of the points $\left(\frac{\|f_d - f_c\|_\mu}{\|f_d\|_\mu}, \overline{d}_{|dc|}(\delta)\right)$ for $\delta = 0.1$ (circles) and $\delta = 1$ (triangles). $S = 2000$, $E = 100$. Left panel: $S_{\min} = 1900$, $S_g = S_{\max} = 2100$. Right panel: $S_{\min} = 1500$, $S_g = S_{\max} = 2500$.

bound increases, as expected. In the right panel it is possible to see that a higher uncertainty on S leads to less informative bounds. Nonetheless, as it will be shown in Figure 7, high levels of uncertainty do not necessarily imply bad estimation performance.

In Figure 5 we focus on the first Monte Carlo run, that well represents the average performance of the proposed estimators. The effectiveness of the estimation strategy (38) is illustrated plotting the true f_μ, the centralized estimate f_c and the distributed estimate f_d ($S_g = 2100$). It is apparent that the quality of the distributed estimator is close to the quality of the centralized estimator.

In Figure 6, considering again the first Monte Carlo run, we show the qualitative dependence of $\overline{d}_{|dc|}(0.1)$ on S and E. As expected, the tightness of the bound generally increases with E and S. This is caused also by the general diminishing of the average of the weighted residuals (51) when S increases.

We then show in Figure 7 the dependency of the quality of the estimates f_d with respect to the accuracy of the guess S_g. The bounds displayed in Figure 4 are different since they are obtained using different values for S_g, related to different levels of knowledge about the true number of sensors S. From (38), one can notice that S_g plays the same role of γ, i.e., S_g is for all purposes a regularization parameter. Hence computing f_d as S_g varies corresponds to obtain the regularization path, see [11, Sec. 16]. Figure 7 shows the relative errors $\frac{\|f_\mu - f_c\|_\mu}{\|f_\mu\|_\mu}$ and

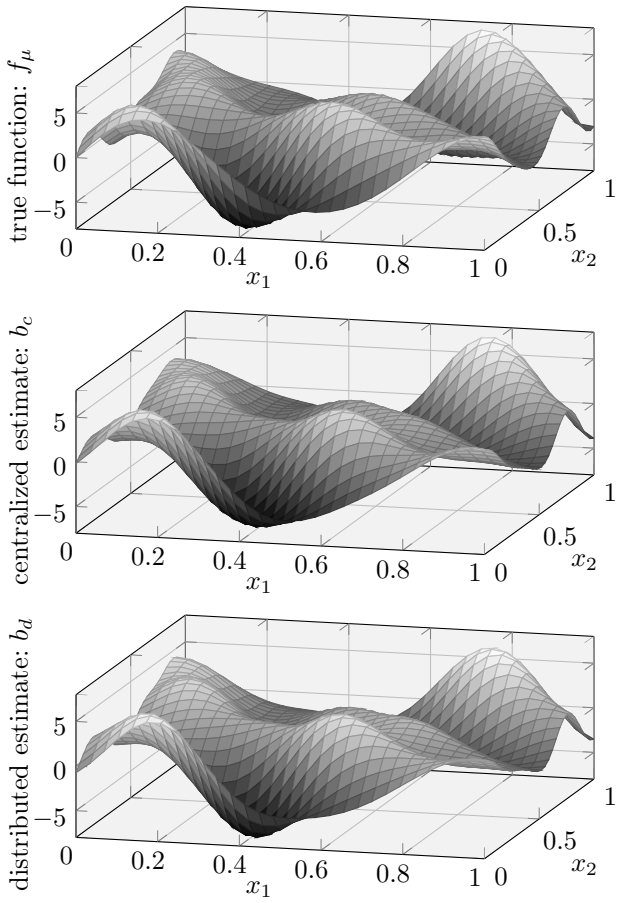

Figure 5. Results of the proposed estimation strategy: true function f_μ (*top*), centralized estimate f_c (*middle*), and distributed estimate f_d (*bottom*), for $S_g = 2100$, $S = 2000$, $E = 100$, $S_{\min} = 1900$ and $S_{\max} = 2100$.

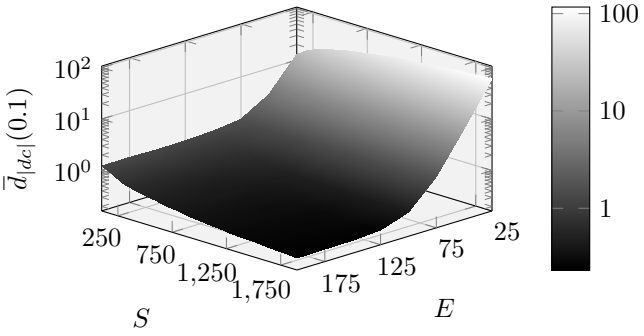

Figure 6. Dependence of $\overline{d}_{|dc|}(0.1)$ on S and E. $S_{\min} = 0.9 \cdot S$, $S_g = S_{\max} = 1.1 \cdot S$.

$\frac{\|f_\mu - f_d\|_\mu}{\|f_\mu\|_\mu}$ for different ratios S_g/S, with $S = 2000$ and $E = 100$, pointing out the robustness of the proposed estimator.

We finally consider the dependency of the accuracy of the estimates with respect to the number of used eigenfunctions E. We show in Figure 8 boxplots summarizing

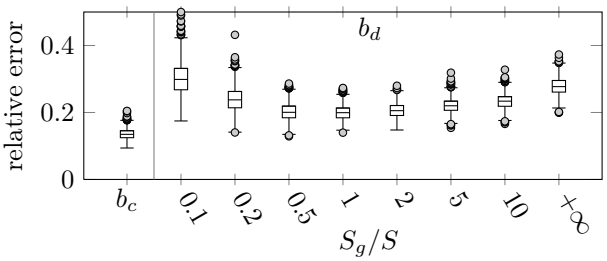

Figure 7. Boxplots relative to the relative errors $\frac{\|f_\mu - f_c\|_\mu}{\|f_\mu\|_\mu}$ (leftmost boxplot) and $\frac{\|f_\mu - f_d\|_\mu}{\|f_\mu\|_\mu}$ (the other boxplots) for different ratios S_g/S, with $S = 2000$ and $E = 100$. $S_g/S = +\infty$ corresponds to a LS solution.

the relative errors $\frac{\|f_\mu - f_d\|_\mu}{\|f_\mu\|_\mu}$ for 1000 independent realizations of f_μ estimated with increasing values of E. As expected, the accuracy tends to increase rapidly with E up to a certain value, and after that the informative content of the succeeding eigenfunctions is negligible. The plot confirms the goodness of the choice $E = 100$.

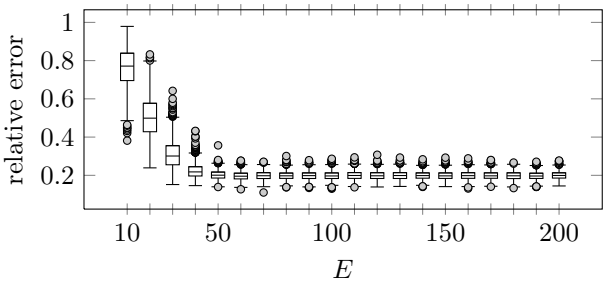

Figure 8. Boxplots relative to the dependency of the accuracy of the estimations with respect to the number of eigenfunctions E. $S = 2000$, $S_g = S_{\max} = 2100$.

8 Conclusions

In this work we have proposed distributed estimators for regression problems in sensor networks without strong topological knowledge and with agents constrained by computational and communication limits. Both parametric and nonparametric scenarios have been considered.

In the parametric context, where the goal is to reconstruct a common finite-dimensional random vector, we proposed a simple distributed estimator and characterized how the estimation accuracy is influenced by the limited knowledge about the actual number of sensors S in the network. We also characterized the estimator performance as a function of the estimate of S, deriving some mild sufficient conditions ensuring the distributed scheme to perform better than the local ones. The performance loss with respect to the optimal centralized estimator has been also quantified.

In the nonparametric scenario we considered the problem of reconstructing a deterministic function from

sparse noisy data collected from the sensors. In this context, the problem is much more difficult since not only S can be uncertain, but also the physical locations where the sensors sample the function can. In addition, one needs to handle the infinite-dimensional nature of the hypothesis space the unknown map is assumed to belong to. We have shown how a distributed version of a Regularization Network can be efficiently computed by the agents just using consensus schemes. More importantly, we have also shown how the agents can compute a certificate of quality on the estimate that accounts for all the uncertainty inherent in the sensor network. To this regard, it is worth also noticing that our analysis permits to interpret the value of S entering the nonparametric estimator as a regularization parameter that trades-off bias and variance by balancing the uncertainty on the number of sensors in the network and the uncertainty on the locations where the function is sampled.

Possible avenues for future research include the use of the proposed results to tune on-line the regularization parameter γ, the analysis of the impact of a finite number of steps in the consensus algorithms on the overall performance, and very importantly, the extension to dynamic scenarios where measurements are sampled sequentially and the unknown function can change over time, possibly using using Kalman filtering or average-tracking strategies.

A Appendix

Proof of Theorem 1: To prove the theorem we check which systems parameters Λ_0, S_g, S, C, σ^2 are s.t.

$$\Lambda_d = \mathrm{var}\left(b - b_d\left(S_g\right)\right) \leq \mathrm{var}\left(b - b_\ell\right) = \Lambda_\ell \ . \qquad \text{(A.1)}$$

Recalling that $V\left(\theta\right) = C\Lambda_0 C^T + \theta I$ it is immediate to verify through the matrix inversion lemma that

$$\begin{aligned} b_d \ &= \ \Lambda_0 C^T \left(\Lambda_0 + \tfrac{\sigma^2}{S_g} I\right)^{-1} \bar{y} \\ &= \ \Lambda_0 C^T V \left(\tfrac{\sigma^2}{S_g}\right)^{-1} \left(Cb + \tfrac{1}{S}\textstyle\sum_{i=1}^{S} \nu_i\right) \ . \end{aligned}$$

Therefore the variance of distributed estimator is given by

$$\begin{aligned} \Lambda_d = \Lambda_0 - 2\Lambda_0 C^T V \left(\frac{\sigma^2}{S_g}\right)^{-1} C\Lambda_0 + \\ + \Lambda_0 C^T V \left(\frac{\sigma^2}{S_g}\right)^{-1} V\left(\frac{\sigma^2}{S}\right) V \left(\frac{\sigma^2}{S_g}\right)^{-1} C\Lambda_0 \ . \end{aligned} \qquad \text{(A.2)}$$

Similarly, for the local estimator we get

$$\Lambda_\ell = \Lambda_0 - \Lambda_0 C^T V (\sigma^2)^{-1} C\Lambda_0 \ . \qquad \text{(A.3)}$$

By substituting the previous two equations into (A.1) and by pre- and post-multiplying by Λ_0^{-1}, we get

$$-2V\left(\frac{\sigma^2}{S_g}\right)^{-1} + V\left(\frac{\sigma^2}{S_g}\right)^{-1} V\left(\frac{\sigma^2}{S}\right) V\left(\frac{\sigma^2}{S_g}\right)^{-1} \leq -V\left(\sigma^2\right)^{-1} \qquad \text{(A.4)}$$

which guarantees $\Lambda_d \leq \Lambda_\ell$. Considering the orthogonal matrix U that diagonalizes $C\Lambda C^T$, i.e., $C\Lambda C^T = UDU^T$, $UU^T = I$, where $D = \mathrm{diag}(d_1, \ldots, d_S)$, we obtain $V\left(\theta\right) = U\left(D + \theta I\right)U^T$. Therefore (A.4) can be written as

$$\begin{aligned} &-2U\left(D + \tfrac{\sigma^2}{S_g} I\right)^{-1} U^T + \\ &+U\left(D + \tfrac{\sigma^2}{S_g} I\right)^{-2} \left(D + \tfrac{\sigma^2}{S} I\right) U^T \leq \\ &\leq -U\left(D + \sigma^2 I\right)^{-1} U^T \end{aligned} \qquad \text{(A.5)}$$

where we also used the fact that diagonal matrices commute. Being U orthogonal, we have that $A \leq 0 \Leftrightarrow UAU^{-1} \leq 0$, so we can remove all the U's from (A.5). Now all the remaining matrices are diagonal, so the condition is satisfied if and only if the inequalities are valid component-wise. Therefore, A.1 is equivalent to

$$\frac{-2}{d_m + \frac{\sigma^2}{S_g}} + \frac{d_m + \frac{\sigma^2}{S}}{\left(d_m + \frac{\sigma^2}{S_g}\right)^2} \leq \frac{-1}{d_m + \sigma^2} \quad m = 1, \ldots, M \qquad \text{(A.6)}$$

that can be rewritten as:

$$p_m\left(S_g\right) := \left(\sigma^2 + (1-S)d_m\right) S_g^2 - 2\sigma^2 S S_g + \sigma^2 S \leq 0 \qquad \text{(A.7)}$$

for all m's. Let us define $\dot{p}_m = \frac{\partial p_m}{\partial S_g}$ and $\ddot{p}_m = \frac{\partial^2 p_m}{\partial S_g^2}$. Now for all m's and d_m's we have that $p_m\left(0\right) = \sigma^2 S > 0$ and $p_m\left(1\right) = \left(1 - S\right)\left(d_m + \sigma^2\right) < \left(1 - S\right)\sigma^2 < 0$ since we are assuming there are at least two sensors. Moreover we also have $\dot{p}_m\left(0\right) = -2\sigma^2 S < 0$ and $\dot{p}_m\left(1\right) = p_m\left(1\right) < 0$. This implies that each $p_m\left(\cdot\right)$ has exactly one root in $(0, 1)$, referred as S'_m, while the other root, referred as S''_m, can be before 0 or after 1 depending on the sign of $\sigma^2 + (1 - S)d_m$, as depicted in Figure A.1.

Now consider a fixed m. Condition (A.7) is assured for $S_g \in [1, S_m)$, where:

$$S_m := \begin{cases} +\infty & \text{if } S''_m < 0 \\ S''_m & \text{otherwise.} \end{cases} \qquad \text{(A.8)}$$

If we define $\overline{S} := \min_m S_m$, condition (A.4) is now assured for $S_g \in \left[1, \overline{S}\right)$.

If $\sigma^2 - (1 - S)d_m \leq 0$ for all m, which is equivalent of saying that $\sigma^2 - (1 - S)d_{\min} \leq 0$, where $d_{\min} := \min_m d_m$, then we have that $\overline{S} = +\infty$. This gives rise to sufficient condition (12)-(a) of the theorem.

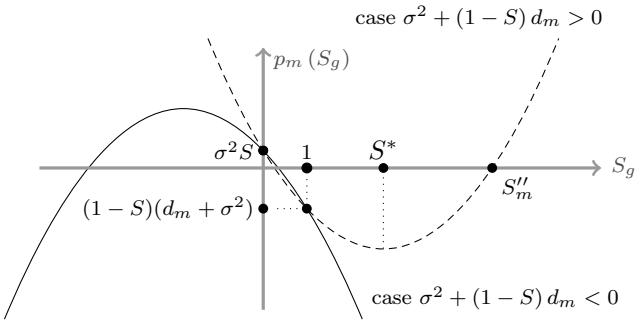

Figure A.1. Example of possible parabolas $p_m\,(S_g)$.

Differently, if $\sigma^2-(1-S)d_{\min} > 0$, let $\overline{m} = \text{argmin}_m d_m$, therefore $\overline{S} = S''_{\overline{m}}$. This point is symmetric to $S'_{\overline{m}}$ with respect to the minimum of the parabola S^*, i.e., $S''_{\overline{m}} = 2S^* - S'_{\overline{m}}$, where

$$S^* = \text{argmin}_{S_g} p_{\overline{m}}(S_g) = \frac{\sigma^2 S}{\sigma^2 + (1-S)d_{\min}} > \frac{\sigma^2 S}{\sigma^2} = S$$

Since $S'_{\overline{m}} < 1$, it follows that $\overline{S} = S''_{\overline{m}} > 2S-1$. This implies that if we restrict $S_g \in [1, 2S-1)$, then (A.4) is satisfied, and this proves the sufficient condition (12)-(b) of the theorem. □

Proof of Theorem 2: Rewriting (8) and (10) as

$$\left(\frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right) b_c = \frac{1}{S}\sum_{i=1}^{S} \frac{C^T y_i}{\sigma^2}$$

$$\left(\frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right) b_d + \left(\frac{1}{S_g} - \frac{1}{S}\right)\Lambda_0^{-1} b_d = \left(\frac{1}{S}\sum_{i=1}^{S} \frac{C^T y_i}{\sigma^2}\right)$$

and subtracting member to member the previous two equations, we obtain

$$\left(\frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)(b_c - b_d) = \left(\frac{1}{S_g} - \frac{1}{S}\right)\Lambda_0^{-1} b_d$$

that implies

$$\frac{\|b_c - b_d\|_2}{\|b_d\|_2} \le \left|\frac{1}{S_g} - \frac{1}{S}\right| \left\| \left(\frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2}\right)^{-1} \Lambda_0^{-1} \right\|_2 .$$

$$\text{Since} \qquad \left|\frac{1}{S_g} - \frac{1}{S}\right| \le \left(\frac{1}{S_{\min}} - \frac{1}{S_{\max}}\right) \tag{A.9}$$

$$\text{and} \qquad \frac{1}{S}\Lambda_0^{-1} + \frac{C^T C}{\sigma^2} \ge \frac{1}{S_{\max}}\Lambda_0^{-1} \tag{A.10}$$

we have

$$\frac{\|b_c - b_d\|_2}{\|b_d\|_2} \le \left(\frac{1}{S_{\min}} - \frac{1}{S_{\max}}\right) \left\| \left(\frac{1}{S_{\max}}\Lambda_0^{-1}\right)^{-1} \Lambda_0^{-1} \right\|_2$$

which is equivalent to (13). □

Proof of Proposition 5: We start noticing that it is possible to associate with $\breve{\mathcal{H}}_K$ the restricted kernel $\breve{K}$ and the relative integral operator $L_{\breve{K},\mu}$ defined respectively by

$$\breve{K}(x, x') := \sum_{e\in\mathcal{I}} \lambda_e \phi_e(x)\, \phi_e(x') \tag{A.11}$$

$$\text{and } L_{\breve{K},\mu}[g](x) := \int_{\mathcal{X}} \breve{K}(x, x')\, g(x')\, d\mu(x') \ . \tag{A.12}$$

Exploiting RKHS theory, see, e.g., [48], one obtains that $\breve{\mathcal{H}}_K$ is exactly the image of $L^{\frac{1}{2}}_{\breve{K},\mu}$ (the square root of $L_{\breve{K},\mu}$) fed with $\mathcal{L}^2(\mu)$, i.e., $\breve{\mathcal{H}}_K = L^{\frac{1}{2}}_{\breve{K},\mu}\left[\mathcal{L}^2(\mu)\right]$.

$$\text{Define } \widehat{f}_\gamma := \arg\min_{f\in\breve{\mathcal{H}}_K} \|f - f_\mu\|^2_\mu + \gamma\|f\|^2_K \tag{A.13}$$

and notice that

$$\widehat{f}_\gamma = \arg\min_{f\in\breve{\mathcal{H}}_K} \left\|f - f_\mu^{\breve{\mathcal{H}}_K}\right\|^2_\mu + \gamma\|f\|^2_K \ . \tag{A.14}$$

In addition, it holds that

$$\begin{aligned}\left\|\widehat{f}_S - f_\mu\right\|_\mu &\le \left\|\widehat{f}_S - \widehat{f}_\gamma\right\|_\mu + \left\|\widehat{f}_\gamma - f_\mu\right\|_\mu \\ &\le \left\|\widehat{f}_S - \widehat{f}_\gamma\right\|_\mu + \left\|\widehat{f}_\gamma - f_\mu^{\breve{\mathcal{H}}_K}\right\|_\mu + \left\|f_\mu^{\breve{\mathcal{H}}_K^\perp}\right\|_\mu .\end{aligned} \tag{A.15}$$

Using theorem 5 in [50], we know that if (26) holds, one has

$$\left\|\widehat{f}_S - \widehat{f}_\gamma\right\|_\mu \le \frac{12\overline{K}\,\overline{Y}\log\left(\frac{4}{\delta}\right)}{\sqrt{\gamma S}} \ . \tag{A.16}$$

In addition, exploiting theorem 3 in [51] and the definition of $L_{\breve{K},\mu}$, it is easy to obtain that, with confidence $1-\delta$, one has

$$\left\|\widehat{f}_\gamma - f_\mu^{\breve{\mathcal{H}}_K}\right\|_\mu \le \sqrt{\gamma}\left\|L^{-\frac{1}{2}}_{\breve{K},\mu}\left[f_\mu^{\breve{\mathcal{H}}_K}\right]\right\|_\mu = \sqrt{\gamma}\left\|f_\mu^{\breve{\mathcal{H}}_K}\right\|_K \ . \tag{A.17}$$

where the last equality exploits $\breve{\mathcal{H}}_K = L^{\frac{1}{2}}_{\breve{K},\mu}\left[\mathcal{L}^2(\mu)\right]$ and the fact that $L^{\frac{1}{2}}_{\breve{K},\mu}$ is an isometric map. The proposition is then proved after simple computations once (A.16), (A.17) and (26) are substituted into (A.15). □

Proof of Proposition 9:

Notice that

$$\frac{\|f_c - f_d\|_\mu}{\|f_d\|_\mu} = \frac{\|b_c - b_d\|_2}{\|b_d\|_2} \le \frac{\|b_c - b_r\|_2}{\|b_d\|_2} + \frac{\|b_r - b_d\|_2}{\|b_d\|_2} \tag{A.18}$$

thus to prove (60) and (61) it is sufficient to characterize $\|b_r - b_d\|_2/\|b_d\|_2$ and $\|b_c - b_r\|_2/\|b_d\|_2$, that will be analyzed separately in the following.

Case $\|b_r - b_d\|_2/\|b_d\|_2$: we start rewriting (36) as $V_r^{-1} b_r = z$ and (38) as $\left(V_r^{-1} + V_d^{-1} - V_r^{-1}\right) b_d = z$. Subtracting the latter to the former we obtain

$$b_r - b_d = V_r \left(V_d^{-1} - V_r^{-1}\right) b_d \tag{A.19}$$

from which it immediately follows that

$$\frac{\|b_d - b_r\|_2}{\|b_d\|_2} = \frac{\left\|V_r \left(V_d^{-1} - V_r^{-1}\right) b_d\right\|_2}{\|b_d\|_2} \ . \tag{A.20}$$

Defining then U_C and U_S by means of (62) and (63), it is immediate to check that $V_d^{-1} - V_r^{-1} = U_S + U_C$,

$$\text{i.e.,}^{10} \qquad \frac{\|b_d - b_r\|_2}{\|b_d\|_2} = \frac{\|V_r U_S b_d + V_r U_C b_d\|_2}{\|b_d\|_2} \ . \tag{A.21}$$

$$\text{Letting} \qquad d_{|dr|} := \frac{\|V_r U_S b_d\|_2}{\|b_d\|_2} + \frac{\|V_r U_C b_d\|_2}{\|b_d\|_2} \tag{A.22}$$

we notice that $d_{|dr|}$ is a random variable since V_r and U_C are random operators. It is then clear that characterizing $d_{|dr|}$ corresponds to characterize the relative error between b_r and b_d. Conditional on z and r_{ave}, b_d is known while the posterior density of the input locations virtually does not vary. This holds in view of the deterministic nature of f_μ, and of the fact that the knowledge of r_{ave} provides a negligible information on the locations visited by the nodes. For this reason [11], up to Monte-Carlo errors $d^*_{|dr|,\text{ave}}$ and $d^*_{|dr|,\text{var}}$ are approximations of $\mathbb{E}\left[d_{|dr|}\right]$ and $\text{var}\left(d_{|dr|}\right)$, respectively.

Now, we prove that the probability density of the random variable $d_{|dr|}$ conditioned on z and r_{ave} has compact support. In fact it is immediate to check that

$$0 \le V_r \le \left(\frac{1}{S_{\max}} \text{diag}\left(\frac{\gamma}{\lambda_e}\right)\right)^{-1} \quad \text{and} \quad U_C \le I \ . \tag{A.23}$$

Moreover, the rank of $\left(C_i^E\right)^T C_i^E$ is one, thus if $\rho(A)$ indicates the spectral radius of A it follows that

$$\rho\left(\left(C_i^E\right)^T C_i^E\right) = \rho\left(C_i^E \left(C_i^E\right)^T\right) = \left\|C_i^E\right\|_2^2 \ . \tag{A.24}$$

[11] Here and in the following, committing a little abuse of notation we drop indications about the fact that expectations and variances are actually conditioned on z and r_{ave}.

Exploiting now the continuity of eigenfunctions on the compact $\mathcal{X}$ we have that

$$\left\|C_i^E\right\|_2^2 \le E \cdot \sup_{x \in \mathcal{X},\ e=1,\dots,E} \left|\phi_e(x)\right|^2 =: \gamma_a < +\infty \tag{A.25}$$

$$\text{thus} \qquad U_C \ge -\frac{1}{S} \sum_{i=1}^{S} \left(C_i^E\right)^T C_i^E \ge -\gamma_a I \ . \tag{A.26}$$

From (A.22), (A.23) and (A.26) it then follows that

$$0 \le d_{|dr|} \le \frac{\left\|S_{\max} \text{diag}\left(\frac{\lambda_e}{\gamma}\right) U_S b_d\right\|_2}{\|b_d\|_2} + \\ + \frac{\left\|S_{\max} \text{diag}\left(\frac{\lambda_e}{\gamma}\right) \max\left(1, \sqrt{\gamma_a}\right) b_d\right\|_2}{\|b_d\|_2} \tag{A.27}$$

that proves that the support of the density of $d_{|dr|}$ is compact.

From (A.27) it follows that $\text{var}\left(d_{|dr|}\right) < +\infty$, and this allows us to use Cantelli's inequality, obtaining

$$\mathbb{P}\left[d_{|dr|} - \mathbb{E}\left[d_{|dr|}\right] \ge \sqrt{\left(\frac{1}{\delta} - 1\right) \text{var}\left(d_{|dr|}\right)}\right] \le \delta \ . \tag{A.28}$$

Rewriting this as

$$\mathbb{P}\left[d_{|dr|} \le \mathbb{E}\left(d_{|dr|}\right) + \sqrt{\left(\frac{1}{\delta} - 1\right) \text{var}\left(d_{|dr|}\right)}\right] \ge 1 - \delta \ , \tag{A.29}$$

considering that $d_{|dr|} \ge \frac{\|f_r - f_d\|_2}{\|f_d\|_2}$, it follows that (60) holds up to Monte Carlo approximations.

Case $\|b_c - b_r\|_2/\|b_d\|_2$: rewriting (36) as

$$\left(\text{diag}\left(\frac{\gamma}{\lambda_e}\right) + \sum_{i=1}^{S} C_i^T C_i\right) b_r + \\ + \left(\sum_{i=1}^{S} \left(C_i^E\right)^T C_i^E - \sum_{i=1}^{S} C_i^T C_i\right) b_r = \\ = \sum_{i=1}^{S} C_i^T y_i - \sum_{i=1}^{S} \left(C_i^{\setminus E}\right)^T y_i \tag{A.30}$$

and (33) as

$$\left(\text{diag}\left(\frac{\gamma}{\lambda_e}\right) + \sum_{i=1}^{S} C_i^T C_i\right) b_c = \sum_{i=1}^{S} C_i^T y_i \tag{A.31}$$

after subtracting (A.31) to (A.30), we obtain

$$\left(\text{diag}\left(\frac{\gamma}{\lambda_e}\right)+\sum_{i=1}^{S}C_i^TC_i\right)(b_c-b_r)= \\ =\left(\sum_{i=1}^{S}\left(C_i^E\right)^TC_i^E-\sum_{i=1}^{S}C_i^TC_i\right)b_r+\sum_{i=1}^{S}\left(C_i^{\setminus E}\right)^Ty_i\ . \tag{A.32}$$

Substituting now each C_i in the right side of (A.32) with $C_i^E+C_i^{\setminus E}$, exploiting the fact that $C_i^{\setminus E}b_r=0$ (where 0 is in $\mathbb{R}^\infty$), and properly collecting the various terms, we obtain

$$b_c-b_r= \\ \left(\text{diag}\left(\frac{\gamma}{\lambda_e}\right)+\sum_{i=1}^{S}C_i^TC_i\right)^{-1}\sum_{i=1}^{S}\left(C_i^{\setminus E}\right)^T(y_i-C_ib_r)\ . \tag{A.33}$$

Since $\text{diag}\left(\frac{\gamma}{\lambda_e}\right)+\sum_{i=1}^{S}C_i^TC_i\geq\text{diag}\left(\frac{\gamma}{\lambda_e}\right)$ (in a matricial positive definite sense), we obtain

$$\|b_c-b_r\|_2\leq\sum_{i=1}^{S}\left\|\text{diag}\left(\frac{\lambda_e}{\gamma}\right)\left(C_i^{\setminus E}\right)^T(y_i-C_ib_r)\right\|_2\ . \tag{A.34}$$

Rewriting $y_i-C_ib_r$ as $y_i-C_i^Eb_d+C_i^Eb_d-C_i^Eb_r$ and using definitions (51), (54) and (A.21) it follows immediately that

$$\begin{aligned}\frac{\|b_c-b_r\|_2}{\|b_d\|_2}\ \leq\ &\sum_{i=1}^{S}\frac{\left\|\text{diag}\left(\frac{\lambda_e}{\gamma}\right)\left(C_i^{\setminus E}\right)^T(y_i-C_ib_d)\right\|_2}{\|b_d\|_2} \\ &+\sum_{i=1}^{S}\left\|\text{diag}\left(\frac{\lambda_e}{\gamma}\right)\left(C_i^{\setminus E}\right)^TC_i^E\right\|_2\frac{\|b_r-b_d\|_2}{\|b_d\|_2} \\ &\leq S_{\max}r_{\text{ave}}+\frac{\|b_r-b_d\|_2}{\|b_d\|_2}S_{\max}s_{\text{ave}}\ .\end{aligned} \tag{A.35}$$

Recalling now (A.18), it immediately follows that

$$\frac{\|b_c-b_d\|_2}{\|b_d\|_2}\leq S_{\max}r_{\text{ave}}+\frac{\|b_r-b_d\|_2}{\|b_d\|_2}\left(1+S_{\max}s_{\text{ave}}\right) \tag{A.36}$$

and thus that if (60) holds, then also (61) does. □

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102 – 114, August 2002.

[2] D. Puccinelli and M. Haenggi, "Wireless sensor networks: applications and challenges of ubiquitous sensing," *IEEE Circuits and Systems Magazine*, vol. 5, no. 3, 2005.

[3] J.-J. Xiao, A. Ribeiro, Z.-Q. Luo, and G. Giannakis, "Distributed compression-estimation using wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 27 – 41, July 2006.

[4] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "Distributed learning in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 56 – 69, July 2006.

[5] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors I: Fundamentals," *Proceedings of the IEEE*, vol. 85, pp. 64 – 63, January 1997.

[6] R. Blum, S. Kassam, and H. V. Poor, "Distributed detection with multiple sensors II: Advanced topics," *Proceedings of the IEEE*, vol. 85, pp. 64 – 79, January 1997.

[7] H. V. Poor, "Competition and Collaboration in Wireless Sensor Networks," in *Sensor Networks*, ser. Signals and Communication Technology, 2009, pp. 3 – 15.

[8] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1997.

[9] P. K. Varshney, *Distributed Detection and Data Fusion*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1996.

[10] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York: Springer, 2001.

[12] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in Ad Hoc WSNs With Noisy Links - Part I: Distributed Estimation of Deterministic Signals," *IEEE Trans. on Signal Processing*, vol. 56, no. 1, pp. 350 – 364, January 2008.

[13] I. D. Schizas and G. B. Giannakis, "Consensus-based distributed estimation of random signals with wireless sensor networks," *40th Asilomar Conference on Signals, Systems and Computers*, pp. 530 – 534, October - November 2006.

[14] G. Mateos, J. A. Bazerque, and G. B. Giannakis, "Distributed Sparse Linear Regression," *IEEE Trans. on Signal Processing*, vol. 58, no. 10, pp. 5262 – 5276, October 2010.

[15] A. Ihler, "Inference in sensor networks: Graphical models and particle methods," Ph.D. dissertation, MIT, June 2005.

[16] V. Delouille, R. Neelamani, and R. Baraniuk, "Robust Distributed Estimation in Sensor Networks using the Embedded Polygons Algorithm," *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, pp. 405 – 413, April 2004.

[17] G. D. Nicolao and G. Ferrari-Trecate, "Consistent identification of NARX models via regularization networks," *IEEE Trans. on Automatic Control*, vol. 44, no. 11, pp. 2045 – 2049, November 1999.

[18] N. Aronszajn, "Theory of reproducing kernels," *Trans. of the American Mathematical Society*, vol. 68, pp. 337 – 404, 1950.

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

[20] B. SchÃ¶lkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[21] D. Varagnolo, G. Pillonetto, and L. Schenato, "Distributed function and time delay estimation using nonparametric techniques," *IEEE Conference on Decision and Control*, pp. 7608 – 7613, December 2009.

[22] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "A Collaborative Training Algorithm for Distributed Learning," *IEEE Trans. on Information Theory*, vol. 55, no. 4, p. , April 2009.

[23] F. Pérez-Cruz and S. R. Kulkarni, "Robust and Low Complexity Distributed Kernel Least Squares Learning in Sensor Networks," *IEEE Signal Processing Letters*, vol. 17, no. 4, April 2010.

[24] P. Honeine, M. Essoloh, C. Richard, and H. Snoussi, "Distributed regression in sensor networks with a reduced-order kernel model," *IEEE Global Telecommunications Conference*, pp. 1 – 5, November - December 2008.

[25] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "Regression in sensor networks: training distributively with alternating projections," *Advanced Signal Processing Algorithms, Architectures, and Implementations XV*, vol. 5910, no. 1, 2005.

[26] C. Guestrin, P. Bodik, R. Thibaux, M. Paskin, and S. Madden, "Distributed Regression: an Efficient Framework for Modeling Sensor Network Data," *Proceedings of the third International Symposium on Information Processing in Sensor Networks (IPSN)*, pp. 1–10, 2004.

[27] K. Yamanishi, "Distributed cooperative Bayesian learning strategies," in *COLT '97: Proceedings of the tenth annual conference on Computational learning theory*. New York, NY, USA: ACM, 1997, pp. 250 – 262.

[28] H. Zheng, S. R. Kulkarni, and H. V. Poor, "Dimensionally Distributed Learning Models and Algorithm," *11th International Conference on Information Fusion*, pp. 1 – 8, June - July 2008.

[29] L. Li, J. A. Chambers, C. G. Lopes, and A. H. Sayed, "Distributed Estimation Over an Adaptive Incremental Network Based on the Affine Projection Algorithm," *IEEE Trans. on Signal Processing*, vol. 58, no. 1, pp. 151 – 164, January 2010.

[30] J. Cortés, "Distributed kriged kalman filter for spatial estimation," *IEEE Transactions on Automatic Control*, vol. 54, no. 12, pp. 2816 –2827, December 2009.

[31] J. Choi, S. Oh, and R. Horowitz, "Distributed learning and cooperative control for multi-agent systems," *Automatica*, vol. 45, no. 12, pp. 2802 – 2814, 2009.

[32] S. Martínez, "Distributed interpolation schemes for field estimation by mobile sensor networks," *IEEE Trans. on Control Systems Technology*, vol. 18, pp. 491 –500, 2010.

[33] Y. Xu, J. Choi, and S. Oh, "Mobile Sensor Network Navigation Using Gaussian Processes With Truncated Observations," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1118–1131, 2011.

[34] G. Pillonetto and B. M. Bell, "Bayes and empirical Bayes semi-blind deconvolution using eigenfunctions of a prior covariance," *Automatica*, vol. 43, no. 10, pp. 1698–1712, 2007.

[35] V. N. Vapnik, *The nature of statistical learning theory*. New York: Springer, 1995.

[36] T. Evgeniou, M. Pontil, and T. Poggio, "Regularization networks and support vector machines," *Advances in Computational Mathematics*, vol. 13, no. 1, pp. 1 – 50, 2000.

[37] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[38] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. on Information Theory/ACM Trans. on Networking*, vol. 52, no. 6, pp. 2508–2530, June 2006.

[39] F. Fagnani and S. Zampieri, "Randomized consensus algorithms over large scale networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 634 – 649, May 2008.

[40] F. Garin and L. Schenato, *Networked Control Systems*, ser. Springer Lecture Notes in Control and Information Sciences. Springer, 2011, ch. A Survey on distributed estimation and control applications using linear consensus algorithms, pp. 75–107.

[41] J. Cortés, "Distributed algorithms for reaching consensus on general functions," *Automatica*, vol. 44, no. 3, pp. 726 – 737, March 2008.

[42] D. Varagnolo, G. Pillonetto, and L. Schenato, "Distributed consensus-based Bayesian estimation: sufficient conditions for performance characterization," in *American Control Conference*, June - July 2010, pp. 3986 – 3991.

[43] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Prentice-Hall, 1979.

[44] A. N. Tikhonov and V. Y. Arsenin, *Solution of Ill-posed Problems*. Wiston, 1977.

[45] K. Yosida, *Functional Analysis*. Springer-Verlag, 1965, vol. 123.

[46] T. Poggio and F. Girosi, "Networks for approximation and learning," *Proc. of the IEEE*, vol. 78, no. 9, September 1990.

[47] G. Wahba, "Spline models for observational data," *SIAM*, 1990.

[48] F. Cucker and S. Smale, "On the mathematical foundations of learning," *Bulletin of the American Mathematical Society*, vol. 39, 2001.

[49] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architecture," *Neural Computation*, vol. 7, pp. 219–269, 1995.

[50] S. Smale and D.-X. Zhou, "Learning Theory Estimates via Integral Operators and Their Approximations ," *Constructive approximation*, vol. 26, pp. 153–172, 2007.

[51] ——, "Shannon sampling II: Connections to learning theory," *Applied and Computational Harmonic Analysis*, vol. 19, pp. 285–302, 2005.

[52] C. Micchelli, Y. Xu, and H. Zhang, "Universal Kernels," *J. of Machine Learning Research*, vol. 7, pp. 2651–2667, 2006.

[53] H. Zhu, C. K. I. Williams, R. Rohwer, and M. Morciniec, "Gaussian Regression and Optimal Finite Dimensional Linear Models," in *Neural Networks and Machine Learning*. Springer-Verlag, 1998.

[54] W. Nef, *Linear Algebra*. McGraw-Hill, 1967.

[55] R. I. Oliveira, "Sums of random Hermitian matrices and an inequality by Rudelson," *Elect. Comm. in Probability*, vol. 15, pp. 203–212, 2010.

[56] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov chain Monte Carlo in Practice*. London: Chapman and Hall, 1996.

[57] C. A. Micchelli and M. Pontil, "Learning the kernel function via regularization," *Journal of Machine Learning Research*, vol. 6, pp. 1099–1125, 2005.