Optimal Sampling Strategies for Adaptive Learning of Graph Signals

Paolo Di Lorenzo[1], Paolo Banelli[1], and Sergio Barbarossa[2]

[1] Department of Engineering, University of Perugia, Via G. Duranti 93, 06125, Perugia, Italy

[2] Department of Information Engineering, Electronics, and Telecommunications, Sapienza University of Rome, Via Eudossiana 18, 00184, Rome, Italy

Email: paolo.dilorenzo@unipg.it, paolo.banelli@unipg.it, sergio.barbarossa@uniroma1.it

***Abstract*—The aim of this paper is to propose optimal sampling strategies for adaptive learning of signals defined over graphs. Introducing a novel least mean square (LMS) estimation strategy with probabilistic sampling, we propose two different methods to select the sampling probability at each node, with the aim of optimizing the sampling rate, or the mean-square performance, while at the same time guaranteeing a prescribed learning rate. The resulting solutions naturally lead to sparse sampling probability vectors that optimize the tradeoff between graph sampling rate, steady-state performance, and learning rate of the LMS algorithm. Numerical simulations validate the proposed approach, and assess the performance of the proposed sampling strategies for adaptive learning of graph signals.**

***Index Terms*—Adaptive LMS estimation, graph signal processing, sampling, successive convex approximation.**

I. Introduction

In the last few years, there was a surge of interest in the development of analysis tools for signals defined over a graph (or graph signals) in view of the many potential applications such as big data, biological networks, transportation networks, sensor networks, [1], [2]. Several analysis methods for graph signals were already proposed in [2], [3]–[5]. For instance, the Graph Fourier Transform (GFT) was defined as the projection of the signal onto the eigenvectors of either the graph Laplacian, see, e.g., [1], [6], [7], or of the adjacency matrix, see, e.g. [2], [8]. A very hot topic in GSP is the development of a *sampling theory* for graph signals, which was initially considered in [6], and later extended in [9], [8], [10], [11], [12]. Then, several reconstruction methods have been proposed, either iterative as in [13], [14], or batch, as in [8], [10], [15]. Recently, adaptive strategies for online reconstruction and learning of graph signals were also proposed in [16]–[18], and paved the way to the development of novel adaptive GSP tools. In particular, reference [16] proposed an LMS estimation strategy for adaptive reconstruction of graph signals from a subset of samples smartly collected over the graph. The method was then extended to the distributed setting in [17]. Finally, in [18], the authors proposed a kernel-based reconstruction framework to handle functions evolving over possibly time-varying topologies, leveraging spatio-temporal dynamics of the observed graph signals.

The work of Paolo Di Lorenzo was supported by the “Fondazione Cassa di Risparmio di Perugia”.

In this paper, we first extend the LMS algorithm of [16] to incorporate a *probabilistic sampling* mechanism, where each node in the graph has an assigned probability to be sampled at each time instant. We also derive a mean-square analysis of the proposed method that illustrates the role played by the sampling strategy on the performance of the LMS algorithm. Based on this analysis, we formulate alternative optimization problems that select the sampling probability at each node in the graph, with the aim of minimizing the overall graph sampling rate (or maximizing the mean-square performance) while imposing learning (or sampling) constraints. Several numerical results are reported to validate the theoretical findings, and illustrate the performance of the proposed strategies.

II. Background

We consider a graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$, with $\mathcal{V}=\{1,2,...,N\}$ denoting the set of nodes and $\mathcal{E}=\{a_{ij}\}_{i,j\in\mathcal{V}}$ the set of weighted edges, such that $a_{ij}>0$ if nodes j and i are connected trough an edge, or $a_{ij}=0$, otherwise. The adjacency matrix $\mathbf{A}$ of a graph is defined as $\mathbf{A}=\{a_{ij}\}, i,j=1,\ldots,N$, and the Laplacian is given by as $\mathbf{L}=\text{diag}\{\mathbf{1}^T\mathbf{A}\}-\mathbf{A}$. In the case of undirected graphs, the symmetric Laplacian matrix is positive semi-definite, with eigendecomposition given by $\mathbf{L}=\mathbf{U}\mathbf{\Lambda}\mathbf{U}^H$, where $\mathbf{U}$ collects all the eigenvectors of $\mathbf{L}$ in its columns, whereas $\mathbf{\Lambda}$ is a diagonal matrix containing the (non-negative) eigenvalues.

A graph signal $\boldsymbol{x}$ is a mapping from the vertex set $\mathcal{V}$ to the set of complex numbers $\mathbb{C}$. The GFT $\boldsymbol{s}$ of a signal $\boldsymbol{x}$ is defined as the projection onto the set of eigenvectors $\mathbf{U}=\{\boldsymbol{u}_i\}_{i=1,\ldots,N}$ of the Laplcian [1], i.e.,

$$\boldsymbol{s}=\mathbf{U}^H\boldsymbol{x}. \tag{1}$$

The GFT has been defined in alternative ways, see, e.g., [1], [2], [8]. In this paper, we keep the formulation general such that one can use the more appropriate definition depending on the particular case, e.g., the approach based on the Laplacian matrix if the graph is undirected, or a GFT operator that handle general directed graphs as, e.g., the one proposed in [2] or [19].

III. Adaptive LMS Estimation of Graph Signals

Let us consider a signal $\boldsymbol{x}^o=\{x_i^o\}_{i=1}^N\in\mathbb{C}^N$ defined over the graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$. We assume that the graph signal is bandlimited, i.e., its spectral content is perfectly localized over

ISBN 978-0-9928626-7-1 © EURASIP 2017

a limited set of frequency indices $\mathcal{F}$. Under the bandlimited assumption, from (1), the graph signal $\boldsymbol{x}^o$ can be written in compact form as:

$$\boldsymbol{x}^o = \mathbf{U}_{\mathcal{F}} \boldsymbol{s}^o, \tag{2}$$

where $\mathbf{U}_{\mathcal{F}} \in \mathbb{C}^{N\times|\mathcal{F}|}$ denoted the subset of columns of matrix $\mathbf{U}$ corresponding to the set of frequency indices $\mathcal{F}$, and $\boldsymbol{s}^o \in \mathbb{C}^{|\mathcal{F}|\times 1}$ is the vector of GFT coefficients of the frequency support of the graph signal $\boldsymbol{x}^o$. Let us assume that streaming and noisy observations of the graph signal are sampled over a time-varying subset of vertices. In such a case, the observation taken at time n can be expressed as:

$$\boldsymbol{y}[n] = \mathbf{D}[n]\,(\boldsymbol{x}^o + \boldsymbol{v}[n]) = \mathbf{D}[n]\mathbf{U}_{\mathcal{F}}\boldsymbol{s}^o + \mathbf{D}[n]\boldsymbol{v}[n] \tag{3}$$

where $\mathbf{D}[n] = \mathrm{diag}\{d_i[n]\}_{i=1}^N \in \mathbb{R}^{N\times N}$, with $d_i[n]$ denoting a random sampling binary coefficient, which is equal to 1 if node i is sampled at time n, and 0 otherwise; and $\boldsymbol{v}[n]$ is zero-mean, white noise with covariance matrix $\mathbf{C}_v$.

The learning task consists in recovering the graph signal $\boldsymbol{x}^o$ (or its GFT $\boldsymbol{s}^o$) from the noisy, streaming, and partial observations $\boldsymbol{y}[n]$ in (3). Following an LMS approach [20], the optimal estimate for $\boldsymbol{s}^o$ can be found as the vector that solves the following optimization problem:

$$\min_{\boldsymbol{s}} \;\; \mathbb{E}\,\|\boldsymbol{y}[n] - \mathbf{D}[n]\mathbf{U}_{\mathcal{F}}\boldsymbol{s}\|^2 \tag{4}$$

where $\mathbb{E}(\cdot)$ denotes the expectation operator. The LMS-type solution proceeds to optimize (4) relying only on instantaneous information and by means of a steepest-descent procedure. Thus, letting $\widehat{\boldsymbol{x}}[n]$ and $\widehat{\boldsymbol{s}}[n]$ be the current estimates of vector $\boldsymbol{x}^o$ and $\boldsymbol{s}^o$, respectively, the LMS algorithm for graph signals evolves as illustrated in Algorithm 1, where $\mu > 0$ is a (sufficiently small) step-size, and we have exploited the fact that $\mathbf{D}[n]$ is an idempotent operator. The learning capabilities of the LMS algorithm in (8) are affected random sampling operator $\mathbf{D}[n]$. Thus, in the sequel, we will show how the design of the sampling strategy $\mathbf{D}[n]$ affects the reconstruction capability, the learning rate, and the steady-state performance of Algorithm 1. Before moving forward, we introduce the following two assumptions.

Assumption 1 (Independent sampling): The sampling process $\{d_i[t]\}$ is temporally and spatially independent. ∎

Assumption 2 (Small step-size): The step-size μ is sufficiently small, i.e., higher-order powers of μ can be neglected. ∎

A. Reconstruction Properties

Assuming stationarity of the sampling and observations random processes $\{d_i[n]\}_{i=1}^N$ and $\{\boldsymbol{y}[n]\}$, the optimal solution $\boldsymbol{s}^o$ of problem (4) is obtained through the normal equations:

$$\mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\boldsymbol{s}^o = \mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\,\mathbb{E}\{\boldsymbol{y}[n]\}, \tag{5}$$

where $\boldsymbol{p} = (p_1, \ldots, p_N)^T \in \mathbb{R}^N$ represents the sampling probability vector, with $p_i = \mathbb{E}\{d_i[n]\}$, $i = 1, \ldots, N$, denoting the probability that node i is sampled at time n. From (5), it is clear that reconstruction of $\boldsymbol{s}^o$ is possible only if the positive (semi)definite matrix $\mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}$ is invertible, i.e., if

$$\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right) > 0, \tag{6}$$

Algorithm 1: LMS on Graphs

Start with random $\widehat{\boldsymbol{s}}[0]$ and $\widehat{\boldsymbol{x}}[0] = \mathbf{U}_{\mathcal{F}}\widehat{\boldsymbol{s}}[0]$. Given a sufficiently small step-size $\mu > 0$, for each time $n > 0$, repeat:

$$\begin{aligned} &\text{S.1)} \quad \widehat{\boldsymbol{s}}[n+1] = \widehat{\boldsymbol{s}}[n] + \mu\,\mathbf{U}_{\mathcal{F}}^H \mathbf{D}[n]\,(\boldsymbol{y}[n] - \widehat{\boldsymbol{x}}[n]) \\ &\text{S.2)} \quad \widehat{\boldsymbol{x}}[n+1] = \mathbf{U}_{\mathcal{F}}\widehat{\boldsymbol{s}}[n+1] \end{aligned} \tag{8}$$

where $\lambda_{\min}(\mathbf{Y})$ is the minimum eigenvalue of matrix $\mathbf{Y}$. Also, let us denote the *expected sampling set* by

$$\overline{\mathcal{S}} = \{i = 1, \ldots, N \,|\, p_i > 0\},$$

i.e., the set of nodes of the graph that are sampled with a probability greater than zero. Thus, a necessary condition to have (6) is that

$$|\overline{\mathcal{S}}| \geq |\mathcal{F}|,$$

i.e., the number of nodes sampled in expectation must be larger than equal to the graph signal bandwidth. Proceeding as in [17], it is possible to prove that matrix $\mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}$ is invertible if

$$\left\|\mathbf{D}_{\overline{\mathcal{S}}_c}\mathbf{U}_{\mathcal{F}}\right\|_2 < 1, \tag{7}$$

where $\overline{\mathcal{S}}_c$ is the complement of the expected sampling set, i.e., $\overline{\mathcal{S}}_c = \{i = 1, \ldots, N \,|\, p_i = 0\}$; and $\mathbf{D}_{\overline{\mathcal{S}}_c} = \mathrm{diag}\{\mathbf{1}_{\overline{\mathcal{S}}_c}\}$, where $\mathbf{1}_{\overline{\mathcal{S}}_c}$ is the set indicator vector, whose i-th entry is equal to one, if $i \in \overline{\mathcal{S}}_c$, or zero otherwise. As shown in [10], condition (7) implies that there are no $\mathcal{F}$-bandlimited signals that are perfectly localized over the set $\overline{\mathcal{S}}_c$. Proceeding as in [10], [16], this condition can be proved to be necessary and sufficient for graph signal reconstruction. However, differently from previous works on sampling of graph signals, see, e.g., [6], [8]–[11], [13], [16], condition (7) depends on the *expected* sampling set. As a consequence of condition (7), the proposed LMS algorithm with probabilistic sampling does not need to collect all the data necessary to reconstruct one-shot the graph signal at each iteration (i.e., the graph signal can be always downsampled at each observation), but can learn acquiring the needed information over time. The only important thing required by condition (7) is that a sufficiently large number of nodes is sampled in *expectation* (i.e., they belong to the expected sampling set $\overline{\mathcal{S}}$).

B. Mean-Square Performance

In the next theorem, we illustrate how the sampling probability vector $\boldsymbol{p}$ affects the mean-square behavior of Algorithm 1.

Theorem 1: Assume model (3) and Assumption 1 hold. Then, for any initial condition, the LMS algorithm(8) is mean-square stable if the step-size μ and the sampling probability vector $\boldsymbol{p}$ are chosen to satisfy (7) and

$$0 < \mu < \frac{2\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)}{\lambda_{\max}^2\left(\mathbf{U}_{\mathcal{F}}^H \,\mathrm{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)}. \tag{9}$$

ISBN 978-0-9928626-7-1 © EURASIP 2017

Then, if condition (9) holds, we have

$$\text{MSD} = \lim_{n\to\infty} \sup_{n} \mathbb{E}\|\widehat{\boldsymbol{x}}[n] - \boldsymbol{x}^o\|^2 \tag{10}$$
$$= \frac{\mu}{2}\operatorname{Tr}\left[\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)^{-1}\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v\mathbf{U}_{\mathcal{F}}\right] + O(\mu^2).$$

Finally, under Assumption 2, the learning rate of Algorithm 1 is well approximated by

$$\alpha = 1 - \mu\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right). \tag{11}$$

Proof. The proof can be found in [21]. ■

Remark: The learning rate $\alpha \in (0,1]$ in (11) determines the convergence speed of $\mathbb{E}\|\widehat{\boldsymbol{x}}[n] - \boldsymbol{x}^o\|^2$ towards its steady-state, i.e., the exponential decay of the transient component. Smaller values of α lead to faster convergence of the algorithm.

IV. Optimal Graph Sampling Strategies

The results of Sec. III-B illustrates how the performance of the LMS algorithm strongly depends on the sampling probability vector $\boldsymbol{p}$ [cf. (10) and (11)]. The goal of this section is to develop optimal probabilistic sampling strategies for adaptive learning of graph signals via the LMS algorithm in (8). In the sequel, exploiting Assumption 2, we neglect the term $O(\mu^2)$ in (10), and (11) well represents the learning rate the LMS algorithm. We consider the two alternative problem formulations, which aim at selecting the sampling probability vector $\boldsymbol{p}$ under different optimization criteria.

A. Minimum sampling rate with learning constraints

The first sampling strategy aims at selecting the probability vector $\boldsymbol{p}$ that minimizes the overall sampling rate over the graph, while guaranteing a target performance of the LMS algorithm in terms of MSD in (10) and of learning rate in (11). The optimization problem can be cast as:

$$\begin{aligned} &\min_{\boldsymbol{p}} \quad \mathbf{1}^T\boldsymbol{p} \\ &\text{subject to} \\ &\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right) \geq \frac{1-\bar{\alpha}}{\mu} \\ &\operatorname{Tr}\left[\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)^{-1}\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v\mathbf{U}_{\mathcal{F}}\right] \leq \frac{2\gamma}{\mu} \\ &\mathbf{0} \leq \boldsymbol{p} \leq \boldsymbol{p}^{\max} \end{aligned} \tag{12}$$

The first constraint imposes that the learning rate of the algorithm is larger than a desired value, i.e., α in (11) is smaller than a target value, say, e.g., $\bar{\alpha} \in (0,1)$. Note that the first constraint on the learning rate also guarantees adaptive signal reconstruction [cf. (6)]. The second constraint guarantees a target mean-square performance, i.e., the MSD in (10) must be less than or equal to a prescribed value, say, e.g., $\gamma > 0$. Finally, the last constraint limits the probability vector to lie in the box $p_i \in [0, p_i^{\max}]$, for all i, with $0 \leq p_i^{\max} \leq 1$ denoting an upper bound on the sampling probability at each node that might depend on external factors such as, e.g., limited energy, processing, and/or communication resources, failures, etc.

Unfortunately, problem (12) is non-convex, due to the presence of the non-convex constraint on the MSD. To handle the non-convexity of (12), one might use successive convex approximation methods with provable convergence guarantees to local optimal solutions of (12) [22]. However, in this paper, we follow a different approach. In particular, under Assumption 2 [i.e., neglecting the terms $O(\mu^2)$], we exploit an upper bound of the MSD function in (10), given by:

$$\text{MSD}(\boldsymbol{p}) \;\leq\; \overline{\text{MSD}}(\boldsymbol{p}) \triangleq \frac{\mu}{2}\,\frac{\operatorname{Tr}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v\mathbf{U}_{\mathcal{F}}\right)}{\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)}, \tag{13}$$

for all $\boldsymbol{p} \in \mathbb{R}^N$. Of course, replacing the MSD function (10) with the bound (13), the second constraint of problem (12) is always satisfied. Thus, exploiting the bound in (13), we formulate a surrogate optimization problem for the selection of the probability vector $\boldsymbol{p}$, which can be cast as:

$$\begin{aligned} &\min_{\boldsymbol{p}} \quad \mathbf{1}^T\boldsymbol{p} \\ &\text{subject to} \\ &\qquad \lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right) \geq \frac{1-\bar{\alpha}}{\mu} \\ &\qquad \frac{\operatorname{Tr}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v\mathbf{U}_{\mathcal{F}}\right)}{\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)} \leq \frac{2\gamma}{\mu} \\ &\qquad \mathbf{0} \leq \boldsymbol{p} \leq \boldsymbol{p}^{\max} \end{aligned} \tag{14}$$

Problem (14) is now a convex optimization problem. Indeed, the second constraint of problem (14) involves the ratio of a convex function over a concave function. Since both functions at numerator and denumerator of (13) are differentiable and positive for all $\boldsymbol{p}$ satisfying the first and third constraint of problem (14), the function is pseudo-convex [23], and all its sub-level sets are convex sets. This argument, coupled with the convexity of the objective function and of the sets defined by the first and third constraints, proves the convexity of the problem (14), whose global solution can be found using efficient numerical tools [24].

B. Minimum MSD with sampling and learning constraints

The second sampling strategy aims at selecting the probability vector $\boldsymbol{p}$ that minimizes the MSD in (10), while imposing that the learning rate of the algorithm is larger than a desired value, and the sampling rate is limited by some budget constraint. The optimization problem can then be cast as:

$$\begin{aligned} &\min_{\boldsymbol{p}} \quad \operatorname{Tr}\left[\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)^{-1}\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v\mathbf{U}_{\mathcal{F}}\right] \\ &\text{s.t.} \quad \boldsymbol{p} \in \mathcal{C} \triangleq \begin{cases} \lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right) \geq \dfrac{1-\bar{\alpha}}{\mu} \\ \mathbf{0} \leq \boldsymbol{p} \leq \boldsymbol{p}^{\max} \\ \mathbf{1}^T\boldsymbol{p} \leq \mathcal{P} \end{cases} \end{aligned} \tag{15}$$

where $\mathcal{P} \in [0, \mathbf{1}^T\boldsymbol{p}^{\max}]$ is the budget on the sampling rate. Problem (15) has a convex feasible set $\mathcal{C}$, but it is non-convex because of the MSD objective function. Again, one might use

Sampling Strategy: Dinkelbach method for Problem

Set $k = 1$. Start with $\boldsymbol{p}[1] \in \mathcal{C}$ and $\beta[1] = \psi(\boldsymbol{p}[1])$. The $k \geq 1$, repeat the following steps:

S.1) $\boldsymbol{p}[k+1] = \underset{\boldsymbol{p}\in\mathcal{C}}{\arg\min}\; h(\boldsymbol{p}, \beta[k])$

S.2) If $h(\boldsymbol{p}[k+1], \beta[k]) = 0$, STOP and $\boldsymbol{p}^* = \boldsymbol{p}[k+1$

otherwise, $\beta[k+1] = \psi(\boldsymbol{p}[k+1])$, $k = k+1$, and go t

---

successive convex approximation methods to find local op solutions of (15) [22]. However, as done before, we e: the upper bound in (13) to formulate a surrogate optimiz problem, which reads as:

$$\min_{\boldsymbol{p}\in\mathcal{C}} \frac{\mathrm{Tr}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v \mathbf{U}_{\mathcal{F}}\right)}{\lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right)}.$$

Problem (16) is a convex/concave fractional program [25], i.e., a problem that involves the minimization of the ratio of a convex function over a concave function, both defined over the convex set $\mathcal{C}$. In particular, as mentioned before, the objective function of (16) is pseudo-convex in $\mathcal{C}$ [23]. As a consequence, any local minimum of problem (16) is also a global minimum [25]. To find a solution of the problem (16), in this paper we consider a method based on the Dinkelbach algorithm [26], which converts the fractional problem (16) into the iterative solution of a sequence of parametric problems as:

$$\min_{\boldsymbol{p}\in\mathcal{C}} \; h(\boldsymbol{p}, \beta) \;=\; f(\boldsymbol{p}) - \beta g(\boldsymbol{p}) \tag{17}$$

with β denoting the free parameter to be selected, and

$$f(\boldsymbol{p}) = \mathrm{Tr}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{C}_v \mathbf{U}_{\mathcal{F}}\right), \tag{18}$$

$$g(\boldsymbol{p}) = \lambda_{\min}\left(\mathbf{U}_{\mathcal{F}}^H \operatorname{diag}(\boldsymbol{p})\mathbf{U}_{\mathcal{F}}\right). \tag{19}$$

Letting $\psi(\boldsymbol{p}) = f(\boldsymbol{p})/g(\boldsymbol{p})$, and noting that $h(\boldsymbol{p}^*, \psi(\boldsymbol{p}^*)) = 0$ at the optimal value $\boldsymbol{p}^*$, the Dinkelbach method proceeds as described in the Sampling Strategy 2, and is guaranteed to converge to global optimal solutions of the approximated problem (16), see, e.g., [25], [26].

In the sequel, we will illustrate numerical results assessing the performance of the proposed LMS algorithm with probabilistic sampling strategies (14) and (16).

V. Numerical Results

Let us consider an application to a real network: the IEEE 118 Bus Test Case, which represents a portion of the American Electric Power System (in the Mid-western US) as of December 1962. The graph is composed of 118 nodes, and its topology is illustrated in Fig. 1. The dynamics of the power generators give rise to smooth graph signals, so that the bandlimited assumption is justified in approximate sense. Thus, we assume that the spectral content of the graph signal is limited to the first ten eigenvectors of the Laplacian matrix of the graph in Fig. 1, i.e., $|\mathcal{F}| = 10$. The observation noise in (3) is zero-mean, Gaussian, with a diagonal covariance matrix,

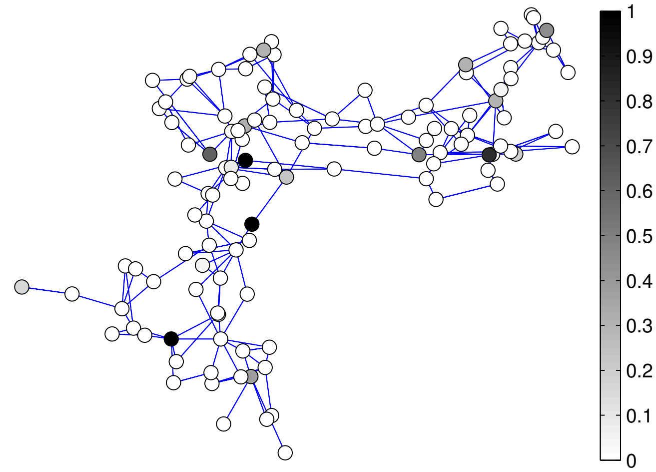

Fig. 1: IEEE 118 bus test case: Graph topology, and optimal sampling probabilities obtained from (16) for $\bar{\alpha} = 0.99$.

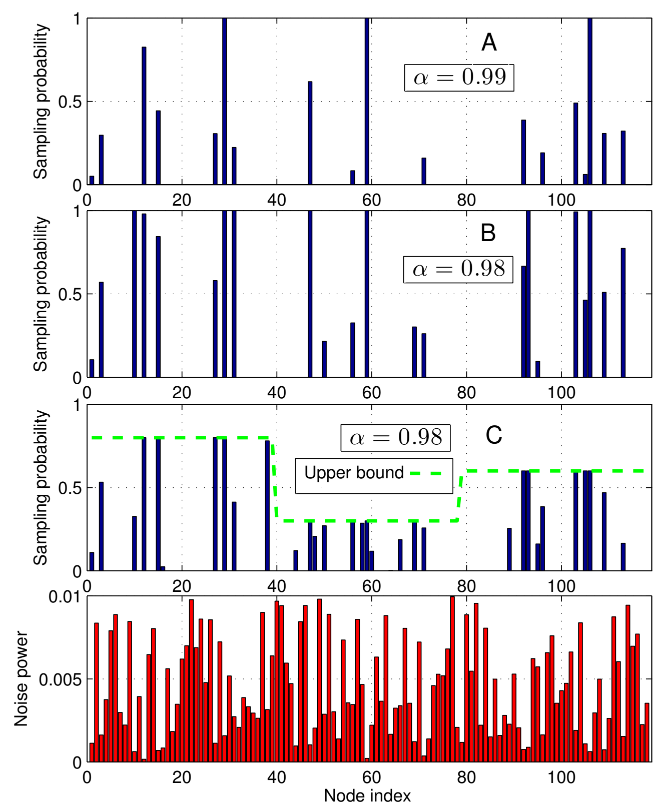

Fig. 2: Optimal probabilities and noise variance, obtained from (16) for different values of $\bar{\alpha}$ and $\boldsymbol{p}^{\max}$.

where each element is illustrated in Fig. 2 (bottom). The other parameters are: $\mu = 0.1$, and $\boldsymbol{p}^{\max} = \mathbf{1}$. An example of optimal probabilistic sampling, obtained solving problem (16) with $\bar{\alpha} = 0.99$ and $\mathcal{P} = 120$, is illustrated in Fig. 1, where the the color (in gray scale) of the vertexes denotes the sampling probability. As we can notice from Fig. 1, the method selects a very sparse probability vector in order to minimize the MSD and guarantee the required learning rate.

As a further example, in Fig. 2 (A , B, and C), we report the optimal probability vector obtained using the Sampling

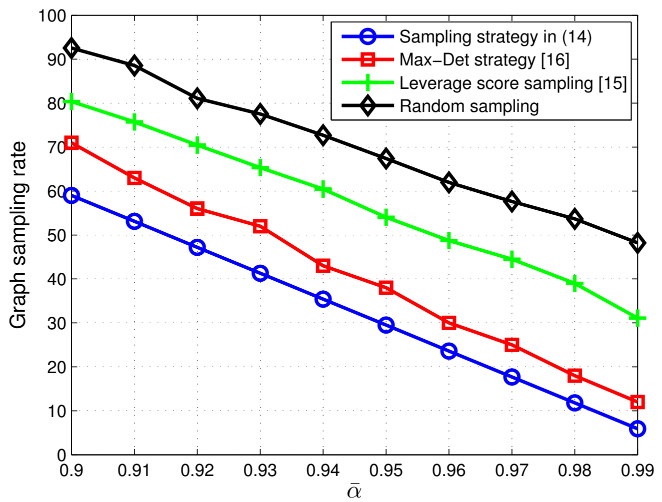

Fig. 3: Graph sampling rate versus $\bar{\alpha}$, for different strategies.

Strategy 2 for different values of $\bar{\alpha}$ (0.99 for case A, 0.98 for B and C) and upper bound vectors $\boldsymbol{p}^{\max}$. In all cases, we have checked that the constraint on the learning rate is attained strictly. From Fig. 2 (A and B), as expected, we notice how the method enlarges the expected sampling set if we require a faster learning rate (i.e., a smaller value of $\bar{\alpha}$), or if there are strict bounds on the probability to sample set of "important" nodes (B and C). Also, from Fig. 2, it is clear how the method avoids to assign large sampling probabilities to nodes having large noise variances, in order to keep the MSD as small as possible, while still guaranteing the target learning rate.

Finally, we compare the sampling strategy in (14) with some established sampling methods for graph signals, namely, the leverage score sampling from [15], the Max-Det greedy strategy from [16], and the (uniformly) random sampling. For each strategy, we keep adding nodes to the sampling set according to the corresponding criterion until the constraints on the learning rate and the MSD in (14) are satisfied. Then, in Fig. 3, we report the behavior of the graph sampling rate versus the learning parameter $\bar{\alpha}$ in (14), obtained using the four aforementioned strategies. The other parameters are: $\mu = 0.1$, $\boldsymbol{p}^{\max} = \mathbf{1}$, and $\gamma = -25$ dB. The results for the the random sampling strategies are averaged over 200 independent simulations. As expected, from Fig. 3, we notice how the graph sampling rate increases at lower values of $\bar{\alpha}$, i.e., increasing the learning rate of the algorithm, for all strategies. Furthermore, we can notice the large gain on the graph sampling rate obtained by the proposed strategy with respect to other methods available in the literature.

VI. Conclusions

In this paper we have introduced a novel LMS strategy for learning graph signals based on a probabilistic sampling mechanism. Then, we have formulated two convex optimization problems to select the sampling probability at each node: the first one aims at minimizing the graph sampling rate while imposing learning constraints, whereas the second strategy optimizes the mean-square performance while constraining the graph sampling rate and the learning rate of the LMS algorithm. Numerical results illustrate the performance of the proposed methods for adaptive learning of graph signals.

References

[1] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Proc. Mag.*, vol. 30, no. 3, pp. 83–98, 2013.

[2] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. on Signal Proc.*, vol. 61, pp. 1644–1656, 2013.

[3] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, 2014.

[4] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2786–2799, 2012.

[5] ——, "Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4673–4685, 2013.

[6] I. Z. Pesenson, "Sampling in Paley-Wiener spaces on combinatorial graphs," *Trans. of the American Mathematical Society*, vol. 360, no. 10, pp. 5603–5627, 2008.

[7] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 3921–3924.

[8] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Trans. on Signal Proc.*, vol. 63, pp. 6510–6523, Dec. 2015.

[9] S. Narang, A. Gadde, and A. Ortega, "Signal processing techniques for interpolation in graph structured data," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 5445–5449.

[10] M. Tsitsvero, S. Barbarossa, and P. Di Lorenzo, "Signals on graphs: Uncertainty principle and sampling," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4845–4860, 2016.

[11] A. G. Marquez, S. Segarra, G. Leus, and A. Ribeiro, "Sampling of graph signals with successive local aggregations," *IEEE Trans. on Signal Process.*, vol. 64, no. 7, pp. 1832–1843.

[12] L. F. Chamon and A. Ribeiro, "Greedy sampling of graph signals," *arXiv preprint arXiv:1704.01223*, 2017.

[13] X. Wang, P. Liu, and Y. Gu, "Local-set-based graph signal reconstruction," *IEEE Trans. on Signal Proc.*, vol. 63, no. 9, pp. 2432–2444, 2015.

[14] S. Segarra, A. G. Marques, G. Leus, and A. Ribeiro, "Reconstruction of graph signals through percolation from seeding nodes," *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4363–4378, 2016.

[15] S. Chen, R. Varma, A. Singh, and J. Kovačević, "Signal recovery on graphs: Fundamental limits of sampling strategies," *IEEE Trans. on Signal and Inf. Proc. over Networks*, vol. 2, no. 4, pp. 539–554, 2016.

[16] P. Di Lorenzo, S. Barbarossa, P. Banelli, and S. Sardellitti, "Adaptive least mean squares estimation of graph signals," *IEEE Trans. on Sig. and Inf. Proc. over Networks*, vol. 2, no. 4, pp. 555–568, 2016.

[17] P. Di Lorenzo, P. Banelli, S. Barbarossa, and S. Sardellitti, "Distributed adaptive learning of graph signals," *To appear in IEEE Transactions on Signal Processing*, 2017.

[18] D. Romero, V. N. Ioannidis, and G. B. Giannakis, "Kernel-based reconstruction of space-time functions on dynamic graphs," *arXiv preprint arXiv:1612.03615*, 2016.

[19] S. Sardellitti, S. Barbarossa, and P. Di Lorenzo, "On the graph fourier transform for directed graphs," *Available at: http://arxiv.org/abs/1601.05972*, 2016.

[20] A. H. Sayed, *Adaptive filters*. John Wiley & Sons, 2011.

[21] P. Di Lorenzo, P. Banelli, and S. Barbarossa, "Adaptive graph signal processing: Algorithms and optimal sampling strategies," *Submitted to IEEE Transactions on Signal Processing*, 2017.

[22] G. Scutari, F. Facchinei, and L. Lampariello, "Parallel and distributed methods for constrained nonconvex optimization part 1: Theory," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1929–1944, 2017.

[23] M. Avriel, W. E. Diewert, S. Schaible, and I. Zang, *Generalized concavity*. SIAM, 2010.

[24] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[25] S. Schaible and J. Shi, "Recent developments in fractional programming: single-ratio and max-min case," *Nonlinear analysis and convex analysis*, pp. 493–506, 2004.

[26] W. Dinkelbach, "On nonlinear fractional programming," *Management science*, vol. 13, no. 7, pp. 492–498, 1967.