

# Efficient Decentralized Nonlinear Approximation via Selective Gossip

Deniz Üstebay<sup>(\*)</sup>, Rui Castro, Michael Rabbat

## Abstract

Recently, gossip algorithms have received much attention from the wireless sensor network community due to their simplicity, scalability and robustness. Motivated by applications such as compression and distributed transform coding, we propose a new gossip algorithm called Selective Gossip. Unlike the traditional randomized gossip which computes the average of scalar values, we run gossip algorithms in parallel on the elements of a vector. The goal is to compute only the entries which are above a defined threshold in magnitude, i.e., significant entries. Nodes adaptively approximate the significant entries while abstaining from calculating the insignificant ones. Consequently, network lifetime and bandwidth are preserved. We show that with the proposed algorithm nodes reach consensus on the values of the significant entries and the indices of insignificant ones. We illustrate the performance of our algorithm with a field estimation application. For regular topologies, selective gossip computes an approximation of the field using the wavelet transform. For irregular network topologies, we construct an orthonormal transform basis using eigenvectors of the graph Laplacian. Using two real sensor network datasets we show substantial communication savings over randomized gossip. We also propose a decentralized adaptive threshold mechanism such that nodes estimate the threshold while approximating the entries of the vector for computing the best  $m$ -term approximation of the data.

## I. INTRODUCTION

With the recent advancements in wireless sensor networks and cyber-physical systems applications, the need for distributed signal processing algorithms increases. The sizes of such networks continue to grow

Deniz Üstebay (\*: corresponding author), and Michael Rabbat are with Electrical and Computer Engineering Department in McGill University, Montréal, QC, Canada. Email: deniz.ustebay@mail.mcgill.ca, michael.rabbat@mcgill.ca

Rui Castro is with the Department of Mathematics and Computer Science, Eindhoven University of Technology - TU/e, The Netherlands. Email: rmcastro@tue.nl (Some of this work was performed while the author was affiliated with Columbia University.)

Portions of this work were presented in [1].

and the network lifetime stays an important constraint. For large networks, collecting and processing data at a fusion center is not ideal since it creates a single point of failure as well as bottlenecks in the network. In many situations the overall communication cost of centralized algorithms, which includes the cost of establishing routing protocols, can be comparable or significantly higher than that of in-network signal processing algorithms, the latter providing robust and scalable solutions. Among such algorithms, randomized gossip is an iterative decentralized computation scheme which is performed via asynchronous information exchanges. In the basic scalar setting, each node in the network has an approximation of the quantity that is computed. Nodes update their approximations based on information exchanged with one-hop neighbors. The updates asymptotically result in consensus, i.e., all nodes converge to the same approximation. As the algorithm is asynchronous and local, there is no requirement of routing or coordination and there is no risk of creating a bottleneck or single point of failure. Furthermore, gossip is scalable and robust to changes in network topology and unreliable communication environments.

This paper describes *selective gossip* which is specifically designed to approximate large vectors of network data. Regular randomized gossip is performed on scalar values whereas in selective gossip nodes gossip on the elements of a vector. Motivated by applications in compression and distributed transform coding, we are interested in gossiping only on the elements which contain significant energy (i.e., elements with absolute values higher than a threshold) in order to conserve energy and bandwidth. However, with the regular gossip algorithm, we cannot determine which elements are significant before actually computing them as gossip is an iterative computation scheme.

Selective gossip solves this problem by adaptively determining which elements are significant and which are insignificant while gossiping. When nodes gossip on vectors of data, they abstain from gossiping on insignificant components of the vector. In particular, at each round of gossip, two neighboring nodes exchange information for components of the vector that at least one of the nodes believes to be significant, i.e., at least one node approximates this coefficient to be higher than the threshold in absolute value. Hence, the components for which both nodes have approximations lower than a threshold are not exchanged or updated. In the long run, few transmissions are spent on insignificant components and network resources are instead used to compute components which contain significant energy. We prove that selective gossip converges asymptotically. In particular, all the nodes in the network reach consensus on the values of the significant components. On the other hand, for the insignificant components, all nodes in the network terminate computation with approximations which are below the threshold in absolute value, and all nodes obtain approximations which are below the threshold after a finite number of iterations. Therefore, all nodes reach a consensus on which components to disregard. We show how selective gossip can be

used for sparse approximation in a field estimation application. It turns out that selective gossip obtains a network-wide approximation with considerably fewer transmissions compared to naïvely gossiping in parallel on all coefficients.

The rest of the paper is organized as follows. We first continue with background and related work on gossip algorithms and distributed compression schemes. In Section II, we formally describe selective gossip in detail and in Section III we prove that selective gossip converges. In Section IV we describe how selective gossip can be used for distributed transform coding, and in Section V we illustrate the performance of selective gossip for a field estimation application. Section VI proposes a variant of selective gossip which eliminates the requirement of a fixed threshold. Finally in Section VII we conclude with remarks and future work.

#### A. Background and Related Work

Distributed consensus, which was first discussed in the seminal work of Tsitsiklis [2], has been identified as a canonical problem in distributed signal processing and control (see, e.g., [3], [4] for surveys). A subproblem in the distributed consensus framework is called average consensus. For a network of  $n$  nodes and each node having a scalar value  $y_i$ , the goal of average consensus is to compute the average  $\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$  at all nodes. The problem formulation is simple yet the solution of this problem is powerful as it can be easily modified to compute any linear function of the network data. Consequently, consensus algorithms have been used in many applications ranging from decentralized compression [5] to localization in sensor networks [6].

Gossip algorithms solve the average consensus problem asymptotically through local information exchanges between neighboring nodes. Randomized gossip is described and analyzed in [7]. At every iteration of randomized gossip, one of the nodes in the network wakes up uniformly at random. This node randomly chooses one of its neighbors, they exchange values, and both nodes take as their new approximation the average of the values they exchanged. Note that the global average is preserved through iterations. Under very mild conditions on how the neighbor is selected, it can be shown that the values of each node converges to the global average. Recently, several variants of randomized gossip which accelerate the speed of convergence, i.e., reduce the number of required transmissions to reach consensus, have been proposed [8]–[12].

We illustrate the utility of selective gossip in a distributed field estimation application. A compressive sensing-based method of field estimation is presented in [5] where random projections of network data are computed and disseminated across the network using randomized gossip. Wang et al. [13] propose

a distributed algorithm for computing sparse random projections of data. These two approaches are useful for exploratory data analysis but are inefficient when one has available a linear transformation that sparsifies the data. There are also methods using lifted wavelets such as [14] and [15]. Both of these methods require forming specialized routes and work well in static topologies and under reliable wireless networking conditions. However, in the case of time-varying topologies or unreliable wireless links, establishing and maintaining routes will require many transmissions and may cause long delays.

Wuhib et al. [16] present a gossip based protocol for detecting global threshold crossings in decentralized real-time monitoring of IP networks. Similar to selective gossip, this algorithm employs a threshold but it is synchronous, their goal is to raise alerts when a global average of network variable is above the threshold (not to accurately compute significant components of a vector), and they assume that all initial values are positive. The study of gossip-like mechanisms is also of interest to sociophysicists who, e.g., develop and study models of opinion dynamics over networks of individuals; see [17] for a recent survey and [18] for related analytical results. Deffuant et al. [19] propose an asynchronous model where each individual has a continuous opinion and meets other individuals randomly. When two individuals meet and their opinions are close enough, they both perform a gossip-like update; otherwise their opinions remain unchanged. This system models social influence as individuals with similar opinions tend to agree. Although this model seems similar to selective gossip, there are a number of important differences. For example, the objective of selective gossip is to reach a form of consensus across the network, whereas opinion dynamics models often exhibit clustering behavior where different subpopulations converge to different opinions.

This paper builds on preliminary work presented in our conference paper [1]. In particular, the novel contributions of this manuscript include a refined and more detailed proof of convergence, a comparison with the decentralized compression scheme of [5], an evaluation on real sensor data from two deployments, and a decentralized mechanism for adaptive estimation of the selective gossip threshold.

## II. PROBLEM FORMULATION AND ALGORITHM

We consider a network of  $n$  nodes and represent the network connectivity with a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ . The vertices,  $\mathcal{V} = \{1, \dots, n\}$  are the nodes, and the edges  $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$  represent direct communication links between two nodes. We assume that the network is connected and the links are symmetric. Each node  $i \in \{1, \dots, n\}$  in the network has an initial vector,  $\mathbf{y}_i \in \mathbb{R}^m$ . Let  $\bar{\mathbf{y}} = \frac{1}{n} \sum_{i=1}^n \mathbf{y}_i$ . Given a threshold

$\tau > 0$ , our goal is to compute an approximation  $\tilde{\mathbf{y}}$  of  $\bar{\mathbf{y}}$  at every node, where

$$\tilde{y}_j = \begin{cases} \bar{y}_j & \text{if } |\bar{y}_j| \geq \tau, \\ 0 & \text{if } |\bar{y}_j| < \tau, \end{cases} \quad (1)$$

and  $\bar{y}_j$  is the  $j$ th component of  $\bar{\mathbf{y}}$ . We refer to the components  $j$  for which  $|\bar{y}_j| \geq \tau$  as significant and all other components as insignificant.

In order to perform this computation, each node maintains a local approximation  $\mathbf{x}_i(k)$  of  $\tilde{\mathbf{y}}$ , and the approximations are updated iteratively with iterations indexed by  $k$ . For  $j \in \{1, \dots, m\}$ , let  $y_{i,j}$  denote the  $j$ th component of  $\mathbf{y}_i$ . Node  $i$  initializes the  $j$ th component of its local approximation to  $x_{i,j}(0) = y_{i,j}$ . At the  $k$ th iteration, a node  $s$  is chosen uniformly at random from  $\{1, \dots, n\}$  (this can be implemented using the asynchronous time model described in [20]), and  $s$  randomly selects a neighbor  $t \in \mathcal{N}_s$  uniformly at random, where  $\mathcal{N}_s = \{v : (s, v) \in \mathcal{E}\}$  is the set of neighbors of  $s$  in  $\mathcal{G}$ . Then  $s$  and  $t$  gossip only on the significant entries of their approximations; i.e., they update components  $j$  for which either  $|x_{s,j}(k-1)| \geq \tau$  or  $|x_{t,j}(k-1)| \geq \tau$  by setting

$$x_{s,j}(k) = x_{t,j}(k) = \frac{1}{2}(x_{s,j}(k-1) + x_{t,j}(k-1)). \quad (2)$$

No change is made to a component  $j$  when both  $|x_{s,j}(k-1)| < \tau$  and  $|x_{t,j}(k-1)| < \tau$ , and these values are not transmitted, with the aim of saving energy. In particular, when  $|x_{i,j}(k-1)| < \tau$ , node  $i$  considers component  $j$  to be insignificant and can later force it to zero when forming its final local approximation to  $\tilde{\mathbf{y}}$ .

The pseudo-code to simulate selective gossip is given in Algorithm 1. Note that variable  $K_{\max}$  is the user defined maximum number of iterations. The pseudo-code presented is referred to as a simulation of selective gossip since the implementation in practice is a bit different, although entirely equivalent. In particular the gossip update (lines 5-13) can be accomplished with three transmissions. First, node  $s$  sends to node  $t$  the values and indices of its significant components. Then,  $t$  transmits the indices and values of its significant components. At this point,  $s$  has the values for both nodes' significant components, but  $t$  does not have the values at  $s$  for the components that are significant only at  $t$ . To address this, node  $s$  makes another transmission with these values and that completes the gossip update.

### III. CONVERGENCE OF SELECTIVE GOSSIP

In this section we study the convergence of selective gossip. First, we prove that selective gossip asymptotically converges to the correct values for significant components. Since there is no coupling between the different components of the vector  $\tilde{\mathbf{y}}$ , we treat each component individually and focus on

**Algorithm 1** : Selective Gossip

---

```

1: Initialize  $\{\mathbf{x}_i(0)\}_{i \in \mathcal{V}} \leftarrow \mathbf{y}_i$ , threshold  $\tau$ 
2: for  $k = 1, \dots, K_{\max}$  do
3:   Select  $s$  uniformly at random from  $\mathcal{V}$ 
4:   Select  $t$  uniformly at random from  $\mathcal{N}_s$ 
5:   for  $j = 1, \dots, m$  do
6:     if  $(|x_{s,j}(k-1)| \geq \tau$  or  $|x_{t,j}(k-1)| \geq \tau)$  then
7:        $x_{s,j}(k) \leftarrow \frac{1}{2}(x_{s,j}(k-1) + x_{t,j}(k-1))$ 
8:        $x_{t,j}(k) \leftarrow \frac{1}{2}(x_{s,j}(k-1) + x_{t,j}(k-1))$ 
9:     else
10:       $x_{s,j}(k) \leftarrow x_{s,j}(k-1)$ 
11:       $x_{t,j}(k) \leftarrow x_{t,j}(k-1)$ 
12:    end if
13:  end for
14:  for all  $i \in \mathcal{V} \setminus \{s, t\}$  do
15:     $x_i(k) \leftarrow x_i(k-1)$ 
16:  end for
17: end for
18: return  $\mathbf{x}_i(K_{\max})$  for all  $i \in \mathcal{V}$ 

```

---

analyzing the behavior of the algorithm for a single scalar component. Without loss of generality, let  $x_i(0)$  denote the initial value for this component at node  $i$ , let  $\bar{x}$  denote the average, and let  $\tau > 0$  be the given threshold. It is well known that, under the assumptions stated above, randomized gossip converges asymptotically to the average consensus [7]. Selective gossip differs from randomized gossip in that, at some iterations, two nodes may choose not to gossip about a particular component, so it will not be updated. Thus, intuitively, to show convergence when  $\bar{x} \geq \tau$  (resp.,  $-\bar{x} \leq \tau$ ) we just need to show that nodes gossip sufficiently often so that eventually they all have  $x_i(k) \geq \tau$  (resp.,  $-x_i(k) \leq \tau$ ); at that point selective gossip is identical to randomized gossip.

To make this argument rigorous we define a potential function,  $S(k) = \sum_{i=1}^n (x_i(k) - \bar{x})^2$ , and demonstrate that it is strictly decreasing in expectation. First we introduce some notation. Let

$$\mathcal{B}(k) = \{v \in \mathcal{V} : |x_v(k)| < \tau \text{ and } |x_u(k)| < \tau \text{ for all } u \in \mathcal{N}_v\}$$

denote the set of nodes which will not gossip at iteration  $k$ , and let  $\tilde{\mathcal{V}}(k) = \mathcal{V} \setminus \mathcal{B}(k)$  be the set of nodes which have non-zero probability of gossiping at iteration  $k$ . Finally, let  $\tilde{\mathcal{G}}(k) = (\tilde{\mathcal{V}}(k), \tilde{\mathcal{E}}(k))$  be the subgraph of  $\mathcal{G}$  induced by  $\tilde{\mathcal{V}}(k)$ . Our convergence proofs will make use of the following lemma.

*Lemma 1:* Let  $\Delta_{\max} = \max_i |\mathcal{N}_i|$  be the maximum degree of  $\mathcal{G}$ . Let  $(a, b) \in \tilde{\mathcal{E}}(k)$  be a pair of neighboring nodes which has non-zero probability of gossiping at iteration  $k$ . Then,

$$\mathbb{E}[S(k+1)|S(k)] \leq S(k) - \frac{1}{n\Delta_{\max}} (x_a(k) - x_b(k))^2. \quad (3)$$

*Proof:* From the definition of the potential function,  $S(k)$ , and of the gossip update (2), it follows that if nodes  $s$  and  $t$  decide to gossip at iteration  $k+1$ , then  $S(k+1) = S(k) - \frac{1}{2}(x_s(k) - x_t(k))^2$ , and so  $S(k+1) \leq S(k)$  with probability 1. Taking the expectation over the random pair of nodes  $(s, t) \in \mathcal{E}$  drawn at iteration  $k$ , we have

$$\mathbb{E}[S(k+1)|S(k)] = \sum_{s \in \mathcal{V}} \sum_{t \in \mathcal{N}_s} \frac{1}{n|\mathcal{N}_s|} \left( S(k) - \frac{1}{2}(x_s(k) - x_t(k))^2 \mathbb{1}_{\{(s,t) \in \tilde{\mathcal{E}}(k)\}} \right) \quad (4)$$

$$= S(k) - \frac{1}{2} \sum_{(s,t) \in \tilde{\mathcal{E}}(k)} \left( \frac{1}{n|\mathcal{N}_s|} + \frac{1}{n|\mathcal{N}_t|} \right) (x_s(k) - x_t(k))^2 \quad (5)$$

$$\leq S(k) - \frac{1}{n\Delta_{\max}} (x_a(k) - x_b(k))^2, \quad (6)$$

where the indicator in the first line enforces the constraint that nodes  $s$  and  $t$  only gossip (and thus decrease the potential function) if neither one is in  $\mathcal{B}(k)$ . Note that, every edge  $(u, v) \in \tilde{\mathcal{E}}(k)$  contributes twice to the double sum in the first line, once with  $s = u$  and once with  $s = v$ . The inequality follows since  $|\mathcal{N}_i| \leq \Delta_{\max}$  for all nodes  $i$ , and we only count the expected decrease in potential due to the particular pair  $(a, b) \in \tilde{\mathcal{E}}(k)$  gossiping.  $\blacksquare$

We are now ready to state our result for the convergence of significant components.

*Theorem 1:* Let  $S(k)$  be defined as above and suppose  $|\bar{x}| \geq \tau$ . Then

$$\mathbb{E}[S(k)|S(0)] \leq \left( 1 - \frac{1}{n^4 \text{diam}(\mathcal{G})^2 \Delta_{\max}} \right)^k S(0), \quad (7)$$

where  $\text{diam}(\mathcal{G})$  is the diameter of the graph.

*Proof:* We assume that  $S(k) > 0$  (otherwise consensus has been attained). We will show that there exists a pair of neighboring nodes  $(a, b) \in \tilde{\mathcal{E}}(k)$  for which  $(x_a(k) - x_b(k))^2 > S(k)/(n^3 \text{diam}(\mathcal{G})^2)$ . Then we will apply Lemma 1 and find that at every iteration we decrease the potential function by at least  $S(k)/(n^4 \text{diam}(\mathcal{G})^2 \Delta_{\max})$  in expectation. In the rest of the proof, we consider the case  $\bar{x} \geq \tau$  (the case  $\bar{x} \leq -\tau$  is analogous).

To begin, we claim that there exists a node  $a$  with  $x_a(k) \geq \bar{x} + \frac{1}{n}\sqrt{\frac{S(k)}{n}}$ . To see this, observe that there exists a node  $i$  for which  $(x_i(k) - \bar{x})^2 \geq S(k)/n$ ; otherwise, we get the contradiction  $\sum_{i=1}^n (x_i(k) - \bar{x})^2 < S(k)$ . If  $x_i(k) > \bar{x}$  then  $x_i(k) \geq \bar{x} + \sqrt{S(k)/n}$  and we take  $i$  to be the node  $a$  we are looking for. If  $x_i(k) < \bar{x}$ , then  $x_i(k) \leq \bar{x} - \sqrt{S(k)/n}$ . Note that

$$\sum_{j \neq i} x_j(k) = n\bar{x} - x_i(k) = (n-1)\bar{x} + \bar{x} - x_i(k),$$

and therefore  $\sum_{j \neq i} x_j(k) \geq (n-1)\bar{x} + \sqrt{S(k)/n}$ . This implies that there exists a node  $a$  such that

$$x_a(k) \geq \frac{1}{n-1} \left( (n-1)\bar{x} + \sqrt{\frac{S(k)}{n}} \right) \geq \bar{x} + \frac{1}{n}\sqrt{\frac{S(k)}{n}}.$$

Now, define  $\mathcal{H} = \{h \in \mathcal{V} : x_h(k) < \bar{x}\}$  and note that  $\mathcal{H}$  is non-empty since  $S(k) > 0$ . Recall that  $x_a(k) > \bar{x}$ , and let  $a = a_1, a_2, \dots, a_r = h$  be a shortest path in  $\mathcal{G}$  from  $a$  to  $\mathcal{H}$  such that  $h \in \mathcal{H}$  and  $a_\ell \notin \mathcal{H}$  for all  $\ell < r$ . Since  $x_a(k) \geq \bar{x} + \frac{1}{n}\sqrt{\frac{S(k)}{n}}$  and  $x_h(k) < \bar{x}$  it follows that there is at least one step  $(a_\ell, a_{\ell+1})$  along this path for which

$$|x_{a_\ell}(k) - x_{a_{\ell+1}}(k)| \geq \frac{1}{rn} \sqrt{\frac{S(k)}{n}}.$$

Moreover, observe that each node  $a_\ell \in \tilde{\mathcal{V}}(k)$  for all  $\ell = 1, \dots, r$  either considers the coefficient to be significant ( $\ell = 1, \dots, r-1$ ) or it has a neighbor who considers the coefficient to be significant ( $r$  has neighbor  $r-1$ ), and thus  $(a_\ell, a_{\ell+1}) \in \tilde{\mathcal{E}}(k)$ . Since  $r \leq \text{diam}(\mathcal{G})$ , it follows from Lemma 1 that

$$\begin{aligned} \mathbb{E}[S(k+1)|S(k)] &\leq S(k) - \frac{S(k)}{n^4 \text{diam}(\mathcal{G})^2 \Delta_{\max}} \\ &= \left( 1 - \frac{1}{n^4 \text{diam}(\mathcal{G})^2 \Delta_{\max}} \right) S(k), \end{aligned}$$

and recursing back to 0 from iteration  $k$  leads to the claim. ■

Theorem 1 shows that when a component is significant, selective gossip will always compute the correct value in expectation. Standard arguments [7] based on Markov's inequality can be applied to this result to show convergence in probability.

Next, we consider insignificant components for the special case of the complete graph<sup>1</sup>. First, observe that once all nodes believe a component is insignificant, all gossiping on that component will cease; i.e., if  $\tilde{\mathcal{V}}(k) = \emptyset$ , then  $\tilde{\mathcal{V}}(t) = \emptyset$  for all  $t \geq k$ . Thus, for insignificant components, with  $|\bar{x}| < \tau$ , we simply aim to show that the approximations  $x_i(k)$  at every node eventually fall below the threshold  $\tau$  in magnitude.

<sup>1</sup>Recall that the complete graph, denoted  $\mathcal{K}_n$ , on  $n$  nodes is the one where all pairs of nodes are connected with an edge.



*Theorem 2:* Let  $\mathcal{G} = \mathcal{K}_n$  be the complete graph. Suppose that  $|\bar{x}| < \tau$  and  $\tau - |\bar{x}| = c > 0$ . If  $S(0) > 0$  and  $|\tilde{\mathcal{V}}(0)| > 0$ , then

$$\Pr(\tilde{\mathcal{V}}(k) \neq \emptyset) \leq \frac{n(n-1)}{(k+1)c^2} S(0). \quad (8)$$

Thus,  $\Pr(\tilde{\mathcal{V}}(k) = \emptyset) \rightarrow 1$  as  $k \rightarrow \infty$ .

*Proof:* First, for any iteration  $k$  with  $|\tilde{\mathcal{V}}(k)| > 0$ , there exists a node  $a \in \tilde{\mathcal{V}}(k)$  such that  $|x_a(k)| \geq \tau$ , and thus  $(x_a(k) - \bar{x})^2 \geq c^2$ . Also, since gossip iterations preserve the average, we have  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i(k)$ , and so there must be a node  $b$  for which  $|x_a(k) - x_b(k)| \geq |x_a(k) - \bar{x}| \geq c$ . Furthermore,  $(a, b) \in \tilde{\mathcal{E}}(k)$  due to our assumption that  $\mathcal{G} = \mathcal{K}_n$  is the complete graph. Let  $T(k)$  be the indicator function for this event, i.e.  $T(k) = \mathbb{1}_{\{\tilde{\mathcal{V}}(k) \neq \emptyset\}}$ . Therefore, by Lemma 1, and since  $\Delta_{\max} = n - 1$  for the complete graph,

$$\mathbb{E}[S(k+1)|S(k), T(k)] \leq S(k) - \frac{c^2}{n(n-1)} T(k). \quad (9)$$

We take expectation of this equation to get

$$\begin{aligned} \mathbb{E}[\mathbb{E}[S(k+1)|S(k), T(k)]] &= \mathbb{E}[S(k+1)] \\ &\leq \mathbb{E}\left[S(k) - \frac{c^2}{n(n-1)} T(k)\right] \\ &= \mathbb{E}[S(k)] - \frac{c^2}{n(n-1)} \mathbb{E}[T(k)]. \end{aligned}$$

Recurring back to 0 yields

$$\begin{aligned} \mathbb{E}[S(k+1)] &\leq S(0) - \sum_{j=0}^k \frac{c^2}{n(n-1)} \mathbb{E}[T(j)] \\ &\leq S(0) - \frac{(k+1)c^2}{n(n-1)} \mathbb{E}[T(k)], \end{aligned}$$

where the second inequality follows since  $\mathbb{E}[T(k)]$  decays monotonically as  $k$  increases. By assumption, we have  $S(0) > 0$  and  $|\tilde{\mathcal{V}}(0)| > 0$ . If, after  $k$  iterations,  $|\tilde{\mathcal{V}}(k)| > 0$ , then  $S(0) - \frac{(k+1)c^2}{n(n-1)} \mathbb{E}[T(k)] \geq 0$ .

This implies

$$\mathbb{E}[T(k)] \leq \frac{n(n-1)}{(k+1)c^2} S(0),$$

which is equivalent to the claim of the theorem. ■

Theorem 2 addresses the case where  $|\bar{x}| < \tau$  only for the complete graph. This approach does not directly extend to general connected topologies. In particular, in the proof of Theorem 2, one cannot guarantee that the nodes  $a$  and  $b$  will be neighbors in a general topology. However, we conjecture that the theorem can be extended to connected topologies by examining a chain of nodes from  $a$  to  $b$  and ensuring that  $S(k)$  decreases substantially after a sufficient number of iterations.

It is also worth noting that the bounds given in Lemma 1 and Theorems 1 and 2 are extremely loose since we only consider the gossiping of one pair of nodes instead of all pairs in  $\tilde{\mathcal{E}}(k)$ , and hence these bounds should not be taken as an indicator of the rate of convergence. In fact, it is easy to see that once all nodes agree the component is significant, selective gossip behaves identically to randomized gossip, and so asymptotically the rates of convergence are the same as reported in [7] for randomized gossip. As illustrated in the simulations presented below, when some components are insignificant, the error decay rate of selective gossip, as a function of the number of scalar values transmitted, is in fact substantially faster than running randomized gossip in parallel for all components.

#### IV. DECENTRALIZED COMPRESSION AND FIELD ESTIMATION

This section illustrates the use of selective gossip in a distributed field estimation application. We assume that node  $i$  locally measures a value  $f_i \in \mathbb{R}$ , and, stacking the signal components into a vector  $\mathbf{f} \in \mathbb{R}^n$ , our goal is to compute an accurate estimate of  $\mathbf{f}$  at every node, where the accuracy of an estimate  $\tilde{\mathbf{f}}$  is measured via mean squared error  $\|\mathbf{f} - \tilde{\mathbf{f}}\|_2^2/n$ .

Transform coding is based on the idea that many natural signals are sparse or compressible under a suitable linear transformation (see, e.g., [21]). That is, although all signal components  $f_i$  may contain non-negligible energy, under a suitable linear transformation, the energy of the signal concentrates in just a few transform coefficients. Let the collection of vectors  $\Psi = \{\boldsymbol{\psi}_j\}_{j=1}^n$  denote an orthonormal basis for  $\mathbb{R}^n$ . Then we can expand the signal  $\mathbf{f}$  in terms of this basis by writing  $\mathbf{f} = \sum_{j=1}^n \beta_j \boldsymbol{\psi}_j$ , where

$$\beta_j = \mathbf{f}^T \boldsymbol{\psi}_j = \sum_{i=1}^n f_i \psi_{j,i}, \quad (10)$$

are transform coefficients. Sorting the coefficients in descending order of magnitude,

$$|\beta_{(1)}| \geq |\beta_{(2)}| \geq \cdots \geq |\beta_{(n)}|, \quad (11)$$

and arranging the basis vectors in corresponding order (so that  $\beta_{(j)} = \mathbf{f}^T \boldsymbol{\psi}_{(j)}$ ), the  $m$ -term non-linear approximation of  $\mathbf{f}$  in the basis  $\Psi$  approximates  $\mathbf{f}$  using the  $m$  transform coefficients with largest magnitude, and can be written as

$$\tilde{\mathbf{f}}^{(m)} = \sum_{j=1}^m \beta_{(j)} \boldsymbol{\psi}_{(j)}. \quad (12)$$

It is common to say that the signal  $\mathbf{f}$  is sparse under the basis  $\Psi$  if  $\tilde{\mathbf{f}}^{(m)} = \mathbf{f}$  for  $m \geq k$  for some constant  $0 < k < n$  (i.e., only  $k$  of the transform coefficients are non-zero). Similarly, one typically says that  $\mathbf{f}$

is compressed under  $\Psi$  if the mean-squared error decays according to a power-law in the number  $m$  of transform coefficients used in the approximation,

$$\frac{1}{n} \|\mathbf{f} - \tilde{\mathbf{f}}^{(m)}\|_2^2 \leq C m^{-2\alpha}, \quad (13)$$

for constants  $C > 0$  and  $\alpha \geq 1$ . The  $m$ -term approximation lies at the heart of the field of nonlinear approximation [22]. Effective compression via transform coding (i.e., sparse approximation with  $k \ll n$  or compression with  $\alpha \gg 1$ ) depends strongly on the class of signals from which  $\mathbf{f}$  is drawn, and the basis  $\Psi$  employed. In this work, we assume that a suitable transform has been identified, and we focus on efficient decentralized computation of the  $m$ -term approximation; we believe that studying appropriate signal classes of signals, and the corresponding transforms, for network data is an important open problem for future work.

When a signal is sparse or compressible under a linear transform, it is possible to obtain a high-fidelity approximation of  $\mathbf{f}$  by recording the locations and magnitudes of the significant or large-magnitude coefficients. Since each transform coefficient is a linear function of the network signal,  $\mathbf{f}$ , the transform coefficients could be computed directly by executing  $n$  gossip algorithms in parallel (one for each coefficient); then the sorting operation (11) could be carried out locally to obtain the  $m$  coefficients with largest-magnitude. Of course, this is highly inefficient if  $m \ll n$ , since gossip transmissions would be used to compute values which are later discarded, and it is desirable to directly compute the  $m$  largest coefficients. The challenge here is that the locations (i.e., indices) of the  $m$  largest coefficients are signal-dependent and are generally not known a priori.

Instead, selective gossip can be used as a decentralized algorithm to adaptively and efficiently compute the  $m$  coefficients with largest magnitude. We assume that node  $i$  has access to its local measurement,  $f_i$ , as well as the  $i$ th coordinate of each basis vector,  $\{\psi_{j,i}\}_{j=1}^n$ . To initialize selective gossip, node  $i$  sets its  $j$ th initial component to  $x_{i,j}(0) = n f_i \psi_{j,i}$ . Then, for fixed  $\tau > 0$ , those coefficients for which  $|\beta_j| \geq \tau$  will be computed asymptotically at every node; also, all nodes will agree on which coefficients have magnitude  $|\beta_j| < \tau$  below the threshold, and should thus be omitted from the approximation. Note that setting the selective gossip threshold  $\tau$  between the magnitudes of the  $m$ th and  $(m+1)$ st largest transform coefficients (i.e.,  $|\beta_{(m)}| > \tau \geq |\beta_{(m+1)}|$ ) will lead to computation of the  $m$ -term approximation. In the description of selective gossip above, we assumed that a threshold  $\tau > 0$  was specified in advance. Setting the threshold correctly to obtain an  $m$ -term approximation for desired  $m$  without knowing the distribution coefficient magnitudes is impractical. We will return to this issue in Section VI, where we describe a scheme for adapting the threshold  $\tau$  online in order to compute a  $m$ -term approximation.

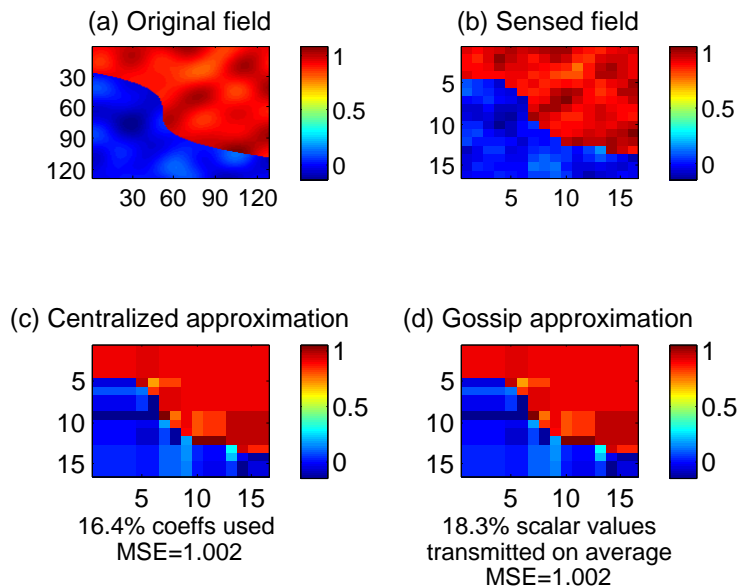


Fig. 1. Original, sensed and approximated field values as color image. For the approximations, threshold value is  $\tau = 0.25$ .

## V. SIMULATION RESULTS

In this section we illustrate the performance of the selective gossip algorithm via simulations. We first consider a grid topology, and using the analogy to images, perform a wavelet transform to estimate a field. Second, we compare the performance of selective gossip to the compressive sensing-based algorithm described in [5]. Third, we use two datasets to show the performance of selective gossip on real data. The topologies are not regular for these datasets hence we use a different transform basis for compression of data. Throughout this section, we count the number of scalar values transmitted as our performance metric instead of the number of gossip iterations. The reason is that the amount of energy expended at each iteration is directly proportional to the number of scalar values transmitted. In a practical implementation, each packet will only be able to carry a small number of coefficients (e.g., the recommended payload for 802.15.4 packets is only 28 bytes), and so large vectors will need to be transmitted as multiple packets. Reducing the number of values transmitted will reduce the total number of packets, and may also shorten the length of the final packet transmitted.

### A. Synthetic data

The field to be estimated is a  $128 \times 128$  discrete sampling of a piecewise smooth field with additive Gaussian noise,  $\mathcal{N}(0, \sigma^2)$ . Figure 1(a) shows this field as a color image. 256 sensor nodes are arranged

with network connectivity forming a  $16 \times 16$  grid. Figure 1(b) is an image generated from the noisy sensor measurements. We use a three-level Haar wavelet basis as the linear transform. Selective gossip is repeated 25 times with different random seeds and the results presented here illustrate the average performance.

Figures 1(c-d) illustrate the results of approximation. For Figure 1(c) the approximations are obtained using the centralized wavelet transform (assuming all the data was gathered at a single location) to compute coefficients and then insignificant coefficients are discarded. Figure 1(d) shows the results of using selective gossip to approximate significant wavelet coefficients. The approximation error is the mean squared error,  $\frac{1}{n} \sum_{i=1}^n \|\tilde{\mathbf{f}}^i - \mathbf{f}\|^2$ , where  $\mathbf{f}$  is the vector of sensor measurements and  $\tilde{\mathbf{f}}^i$  is the reconstructed field using only significant coefficients approximated by node  $i$ . Centralized approximation uses 16.4% of the coefficients to reach the MSE value of 1.002. Selective gossip provides the same MSE value with transmitting 18.3% of scalar values on average where 100% corresponds to transmission of every coefficient approximation, i.e.,  $\tau = 0$ .

Varying  $\tau$  changes the approximation quality. Figure 2 plots mean squared error versus number of gossiped coefficients for different values of the threshold,  $\tau$ . First, Figure 2(a) accounts errors due to both approximation (thresholding coefficients) and gossip,  $\frac{1}{n} \sum_{i=1}^n \|\tilde{\beta}_i - \beta\|^2$ , where  $\beta = \Phi^T \mathbf{f}$  is the vector of true wavelet coefficients and  $\tilde{\beta}_i$  is the vector of approximated coefficients at node  $i$ . The selective gossip curves level off when gossip has effectively converged, and all remaining error is only due to approximation. Figure 2(b) shows the error due to gossip only,  $\frac{1}{n} \sum_{i=1}^n \|\tilde{\beta}_i - \tilde{\beta}\|^2$  where  $\tilde{\beta}$  is the thresholded version of  $\beta$ , as if computed in a centralized fashion, and  $\tilde{\beta}_i$  is the gossip approximation of  $\tilde{\beta}$  at node  $i$ . The error is calculated using thresholded coefficients instead of true coefficients hence error due to approximation is ignored. As expected, higher values of  $\tau$  result in higher approximation error, but with fewer transmitted values.

In Figure 3, we plot the mean squared error due to approximation and gossip versus number of gossip iterations instead of number of scalar values transmitted. We observe that selective gossip requires more iterations to get to a particular MSE value compared to running randomized gossip sessions in parallel, since nodes running selective gossip sometimes do not update significant coefficients. This figure illustrates the trade off between energy-savings and latency which selective gossip provides; that is, selective gossip saves energy by not transmitting insignificant coefficients, but this results in an increased delay in computation since it takes some iterations to determine which coefficients are significant and insignificant.

Figure 4 illustrates how the threshold  $\tau$  influences the number of transmissions invested per coefficient,

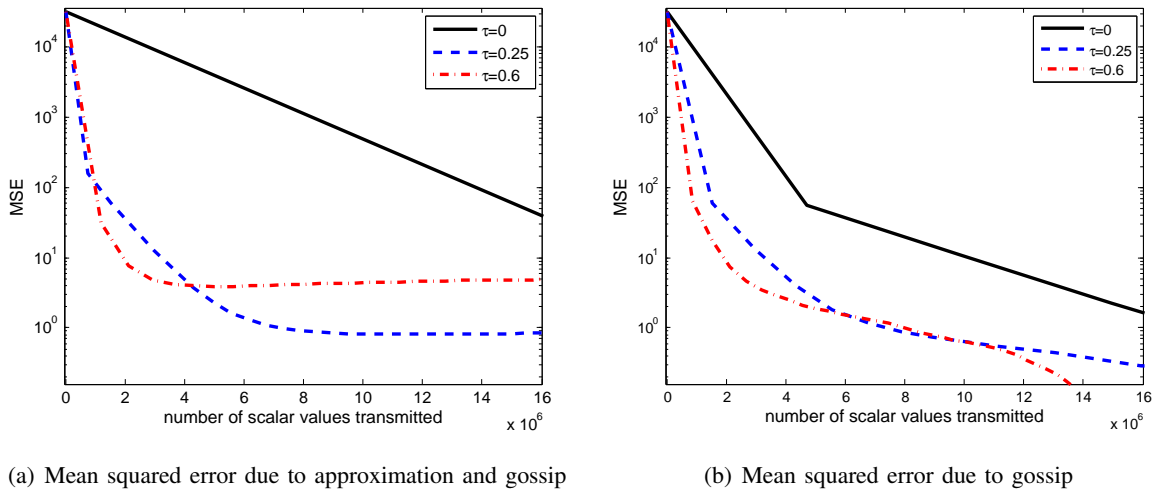


Fig. 2. A comparison of different values of threshold,  $\tau$ . Results are averaged over 25 runs of the algorithm.

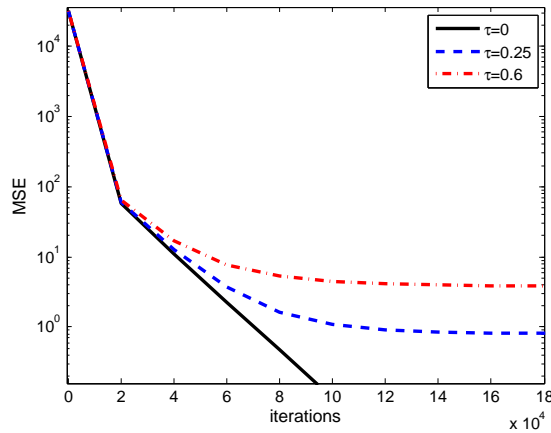


Fig. 3. Mean squared error due to approximation and gossip vs number of gossip iterations, for different values of threshold,  $\tau$ . Results are averaged over 25 runs of the algorithm.

for significant and insignificant coefficients. The top panel shows the original wavelet coefficients in absolute value, sorted in descending order. The panel below shows the number of scalar values transmitted for each coefficient, where the order of indexing is the same as the sorting above. The two curves shown on the bottom panel correspond to two different thresholds,  $\tau = 0.25$  and  $0.6$ . To obtain these curves, we count the number of scalar values transmitted for each coefficient until all nodes agree on the significance or insignificance of that coefficient, namely until the time that the network has finished the selection of this coefficient. If it is an insignificant coefficient, then the selection corresponds to the time when all nodes stop gossiping on this coefficient. On the other hand, if all nodes agree that a coefficient is

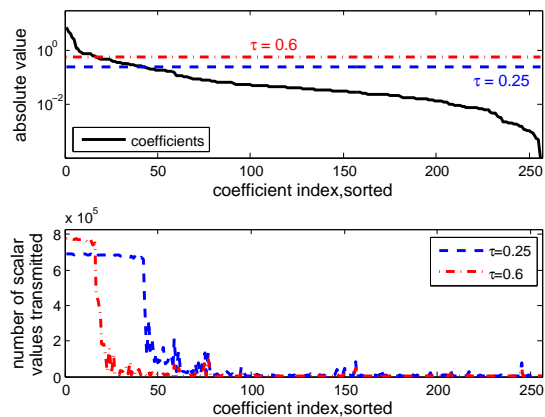


Fig. 4. Coefficients and corresponding energy requirements. Top: original wavelet coefficients shown with the threshold levels. Bottom: For two values of threshold, number of scalar values transmitted for selection of each coefficient.

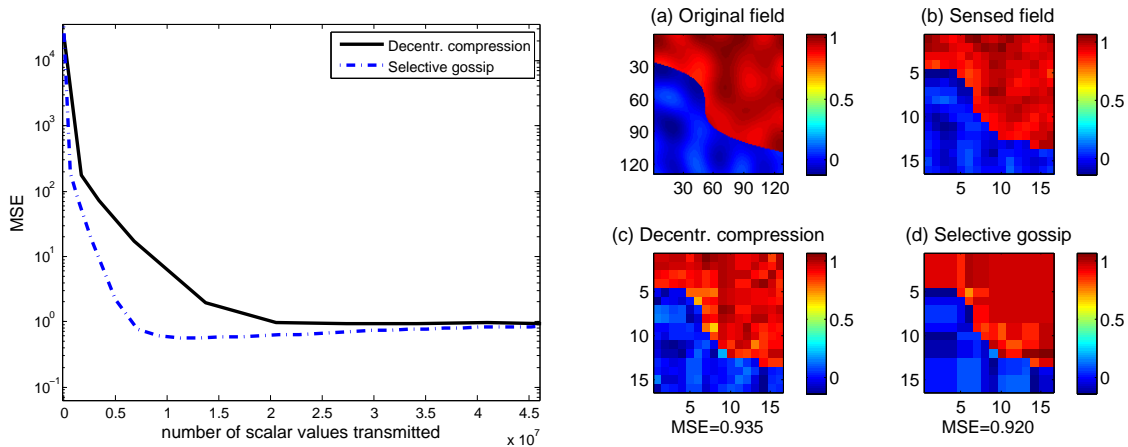


Fig. 5. A comparison of selective gossip and decentralized compression–predistribution. Left panel: Mean squared error versus number of scalar values transmitted. Right panel: Original, sensed fields and approximated fields by decentralized compression and selective gossip.

significant, then they continue gossiping on it until the maximum number of iterations is reached. We observe that selective gossip automatically determines which coefficients are insignificant, and spends a minimal number of transmissions on these coefficients.

### B. Comparison with decentralized compression and predistribution

We compare the performance of selective gossip to the performance of the decentralized compression and predistribution via randomized gossiping [5]. Both of these algorithms utilize randomized gossip to compute sparse approximations of data over a network. The compressed sensing approach of [5] uses

randomized gossip to compute and distribute random projections of the network data. As projections are random, it is not required to identify a sparsifying basis selection prior to the distribution of the projections. In this approach, the computational complexity is pushed to the end user, i.e., the end user needs to know the basis in which the original signal is compressible in order to solve a minimization problem for the reconstruction of the signal. On the other hand, selective gossip requires each node to know one row of the transform matrix prior to executing selective gossip iterations.

The comparison of the two algorithms is carried out using the simulation setting of Section V-A. We investigate the necessary amount of communication for these two algorithms to reach the same MSE value. Gossip is run 25 times for each of the algorithms and the results presented here show the average performance. To reach the same MSE value, we choose a threshold value of  $\tau = 0.25$  for selective gossip and the number of random projections is chosen as 195 for decentralized compression. Figure 5 illustrates that selective gossip is more efficient than decentralized compression in terms of number of scalar values transmitted. Note that the sparsity of the reconstructed signals are not same for the two algorithms. Selective gossip yields an approximation which has 16.4% nonzero elements, whereas decentralized compression reaches the same error value with a less sparse result of 57.8% nonzero elements.

### C. Real sensor network data and general topologies

In this section we investigate the performance of selective gossip on irregular topologies. For regular topologies such as chains or grids we can adopt bases typically used in signal and image processing applications. For general graphs however we need bases adapted to the network topology. One of the transforms for irregular sampled signals is diffusion wavelets [23] which provides multiscale analysis on graphs and manifolds. However this method is proposed for systems with large number of nodes and does not apply to the datasets we work on. Here we use the eigenvectors of the graph Laplacian which provides an orthonormal basis for signals supported on the graph. The graph Laplacian matrix  $L$  for the graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is defined as follows, [24],

$$L_{ij} = \begin{cases} -1 & \text{if } (i, j) \in \mathcal{E} \\ \text{deg}(i) & \text{if } i = j \\ 0 & \text{otherwise,} \end{cases} \quad (14)$$

where  $\text{deg}(i)$  is the degree of vertex  $i$ , i.e., the number of its neighbors. Decentralized computation of eigenvectors of  $L$  can be carried out using, e.g., the scheme of [25].



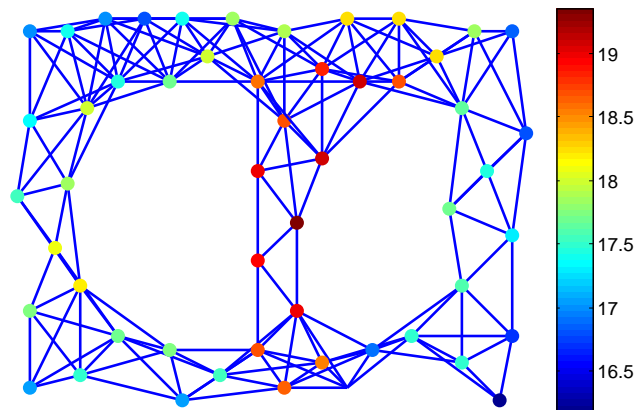


Fig. 6. Network topology, formed based on the distances between the sensors of the Intel Lab data set. The temperature measurements at nodes are indicated with the color coding.

First we used selective gossip on the Intel Lab dataset [26]. In this dataset, 53 sensors are spread in the Intel lab, and they gather measurements of humidity, temperature, light and voltage values once every 31 seconds. We chose to do averaging over temperature measurements (in degrees Celsius). At a single time instant not all nodes have measurements, hence we selected a time interval in which each node has a measurement. Note that the time interval is short enough so that temperature values are nearly constant at all nodes. We took the mean of temperature measurements over this interval at every node and use that as temperature value input to selective gossip. Furthermore, instead of using the network connectivity given in the dataset, we formed a topology based on the distance between sensors such that nodes close to each other are connected. The reason is that nodes which are geographically close to each other are more likely to have similar measurements and hence the signal is smooth on the graph. The constructed graph topology is shown in Figure 6. This figure also shows the temperature readings that are chosen as gossip values.

The network topology of the Intel Lab dataset is not regular and so we cannot use the basis we have previously used for grid topology. Since the temperature is likely a smooth function it can be represented accurately (i.e., sparsely) in a Laplacian eigenvector basis. Hence we construct an orthonormal basis using the eigenvectors of the graph Laplacian and provide simulation results for selective gossip. Figure 7 shows that the basis that we have chosen is actually sparsifying the data, and with a threshold of  $\tau = 1.25$  only 5 coefficients are significant out of a total of 53. The simulation is repeated 50 times with different random seeds and Figure 8 illustrates the average result over 50 runs for the reconstructed measurements. Here worst case reconstruction means that we use the transform coding coefficients of the node with the

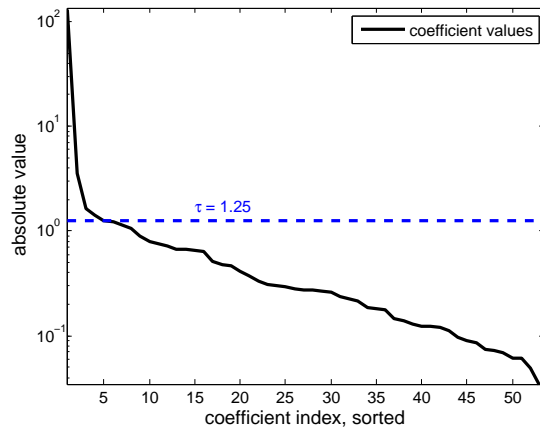


Fig. 7. Transform coefficients for Intel Lab data using the basis constructed from the eigenvectors of the graph Laplacian shown with the threshold value  $\tau = 1.25$ .

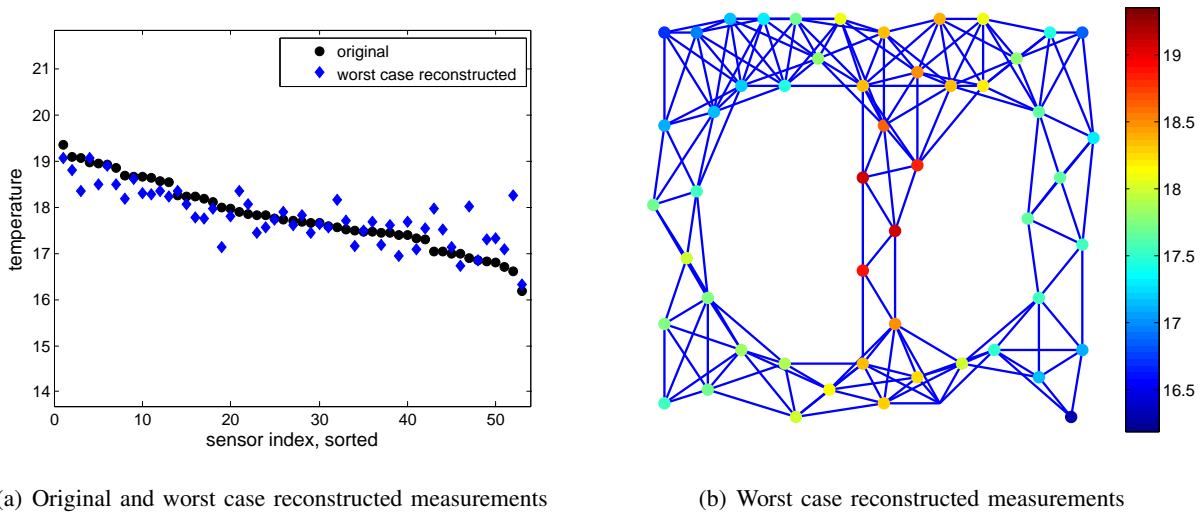


Fig. 8. Comparison of original and worst case reconstructed temperature measurements from Intel Lab data for selective gossip, threshold value  $\tau = 1.25$ .

highest reconstruction error. Observe that even the worst case reconstructed signal which is shown in Figure 8(b) is a good approximation of the original signal shown in Figure 6. Furthermore, we chose more time intervals in the Intel Lab dataset to see the performance over the average of these time intervals. Figure 9 illustrates the MSE performance averaged for 20 time intervals and 50 runs for each interval. Selective gossip is faster compared to running randomized gossip sessions in parallel which is equivalent to having  $\tau = 0$ .

Next, we investigate the performance of selective gossip on data from California Irrigation Management

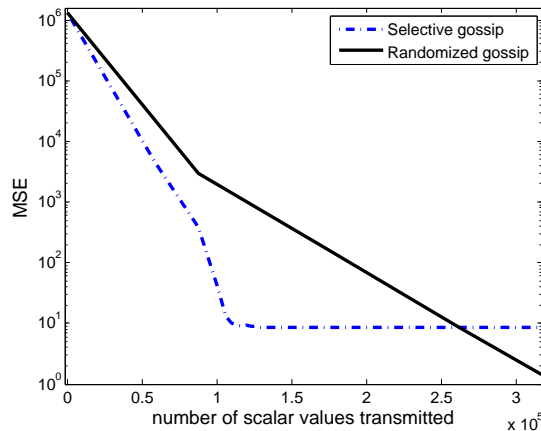


Fig. 9. A comparison of selective gossip and randomized gossip for Intel Lab data, averaged over 20 different time intervals and 50 runs of the algorithm per interval.

Information System (CIMIS) [27]. This dataset is generated by more than 100 automated weather stations in the state of California. The weather stations are equipped with sensors which measure solar radiation, temperature and wind speed every hour. We used the air temperature readings for 24 hours, where each hour corresponds to a time instant in the dataset, to illustrate the field estimation of selective gossip.

The signal that we use is measured by 121 sensors. In the original setting of CIMIS, the sensors send the measurements to a fusion center. Here since we have a distributed scheme we assume a communication network of the sensors. As we have done for Intel Lab dataset, we form a topology by connecting nodes which are close to each other. The resulting network is shown in Figure 10. The signal we first consider is the temperature readings of one hour and is shown with color coding on the topology. Note that the temperature readings of the CIMIS dataset have a much greater dynamic range than Intel Lab dataset temperature values.

Since the topology of the CIMIS network is not regular, again we construct an orthonormal basis from the eigenvectors of the graph Laplacian. Figure 11 shows the resulting transform coefficients as well as the threshold value. We observe that this basis is successful at sparsifying the CIMIS signal and with the threshold of  $\tau = 5$ , only 10 coefficients are significant from a total of 121. In Figure 12, we show the original and reconstructed measurements after simulating 50 runs of selective gossip. Again, worst case reconstruction means that we use the transform coding coefficients from the node with the highest reconstruction error. We can also observe the approximation quality by comparing the original signal in Figure 10 to the worst case reconstruction result in Figure 12(b). Figure 13 illustrates the gain of using selective gossip to distribute an approximation of the data instead of running parallel randomized gossip

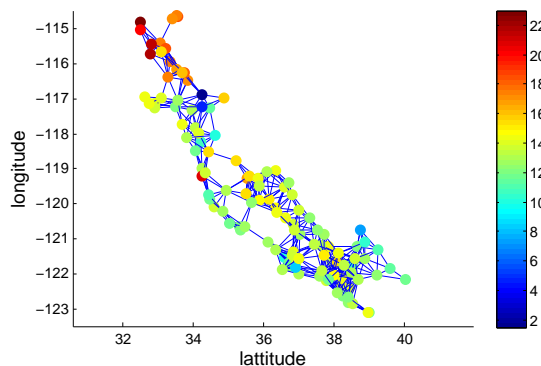


Fig. 10. Network topology, constructed based on the sensor distances of the CIMIS data set. The temperature measurements at nodes are indicated with the color coding.

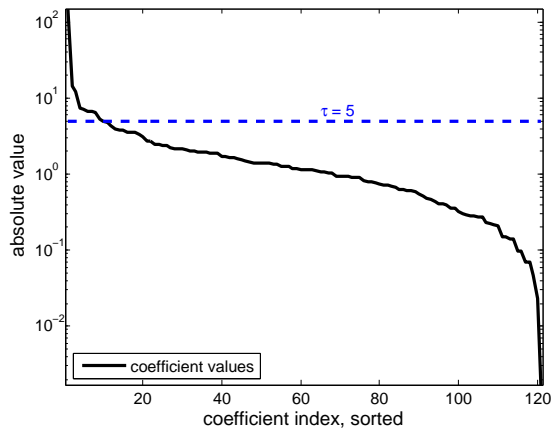
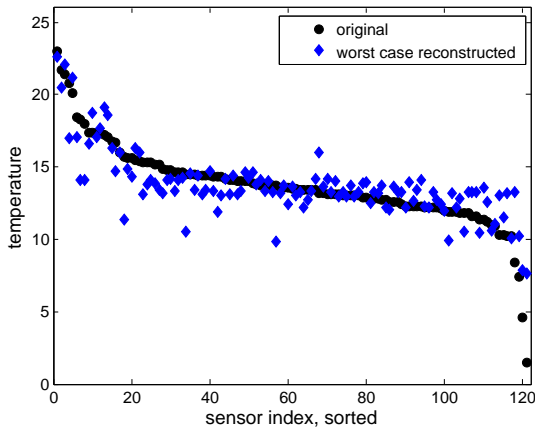


Fig. 11. Transform coefficients for CIMIS data using the basis constructed from the eigenvectors of the graph Laplacian and the threshold value  $\tau = 5$ .

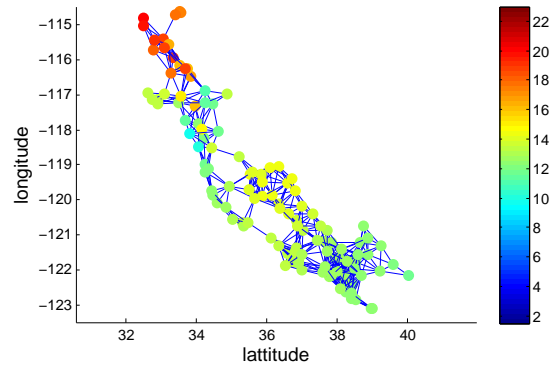
sessions for the whole data. The result shows the performance averaged for 24 hours and 25 runs per hour.

The results obtained for the Intel Lab and the CIMIS datasets are similar for selective gossip although the two topologies are very different from each other as shown in Figures 6 and 10. We can conclude that the choice of basis is successful for these two topologies and the temperature measurements. Furthermore, the performance of selective gossip compared to randomized gossip is better in terms of the number of scalar values transmitted, i.e., the energy spent for the estimation of the field.

Note that, throughout this section, we only considered static networks with lossless links. This helped



(a) Original and worst case reconstructed measurements



(b) Worst case reconstructed measurements

Fig. 12. Comparison of original and worst case reconstructed temperature measurements from CIMIS data for selective gossip, threshold value  $\tau = 5$ .

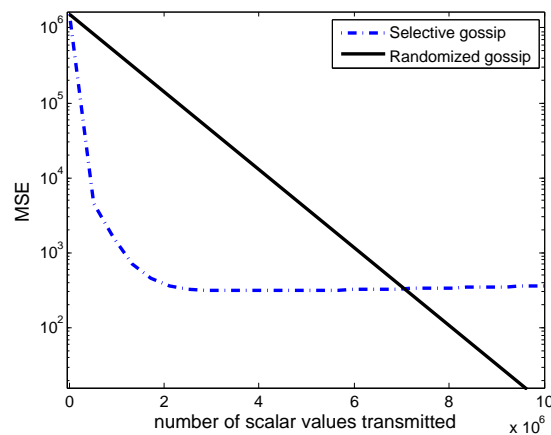


Fig. 13. A comparison of selective gossip and randomized gossip for CIMIS data, averaged over 24 hours and 25 runs of the algorithm per hour.

us illustrate the performance of selective gossip in a controlled environment. However we know that gossip algorithms in general are robust to changing topologies and unreliable networking conditions, as they only depend local information exchanges and routing is not required. Hence, the results presented here can be shown for more general scenarios.

## VI. ADAPTIVE THRESHOLD MECHANISM

Until now, we used a fixed preset threshold,  $\tau$ , to determine the significance of the transform coefficients that are computed by gossip. However in a sensor network setting having a fixed threshold is not practical

as we do not have accurate prior knowledge of coefficient distribution. In this section we describe an adaptive threshold mechanism. Nodes find the appropriate threshold in a decentralized way, without any dependence on the signal or the transform that is used.

Instead of setting a threshold value before aggregating the network data, we would like to have the nodes reach consensus on a preset approximation level. The preset approximation level can be defined by specifying the number of terms  $m$  to use in a best  $m$ -term approximation so that the quality of approximation is chosen by the user, regardless of the signal to be approximated and the transform basis. Note that the selective gossip algorithm described above does not directly compute an  $m$ -term approximation. One could imagine modifying the algorithm so that, rather than gossiping on components with magnitude greater than  $\tau$ , nodes gossip on the union of their  $m$  coefficients with largest magnitude. However, for this modification, there will be cases where the algorithm will no longer produce the correct result. For example, it can happen that a particular coefficient is significant, but the initial values  $|x_i(0)|$  are small at all nodes, in which case the network will never gossip on this component, and all nodes will wrongly consider it insignificant.

Unlike the original selective gossip algorithm which has a pre-defined and fixed threshold which is identical at every node, in the proposed decentralized adaptive mechanism every node keeps an estimate of the threshold as well as the approximations of the coefficients. Initially, the threshold at each node is set to some high value and the goal is to reach the desired best  $m$ -term approximation level at every node by adaptively modifying this threshold. During selective gossip iterations, each node checks its approximation quality. If its current threshold value provides fewer than  $m$  significant coefficients, the node decreases the threshold value. If the node has more than  $m$  significant coefficients, the threshold value is increased at that node. Otherwise, if the node approximation is already at  $m$  then the threshold remains unchanged.

Formally, each node  $i$  has a threshold estimate  $\tau_i(k)$  at time  $k$ . Let  $T_i(k)$  be the number of significant coefficients of node  $i$  at time  $k$ , i.e.  $T_i(k) = \sum_{j=1}^n \mathbb{1}_{\{|x_{ij}(k)| \geq \tau_i(k)\}}$ . The initialization of threshold estimates insures that every node has one significant coefficient at time  $k = 0$ :  $\tau_i(0) = \max_j |x_{ij}(0)|$ . Then nodes update their threshold according to the following rule

$$\tau_i(k+1) = \begin{cases} (1 + c_1)\tau_i(k) & T_i(k) > m \\ (1 - c_2)\tau_i(k) & T_i(k) < m \\ \tau_i(k) & T_i(k) = m \end{cases} \quad (15)$$

where  $c_1, c_2 > 0$  are some constants. Note that we choose  $c_1 \neq c_2$  as having  $c_1 = c_2$  cause oscillations

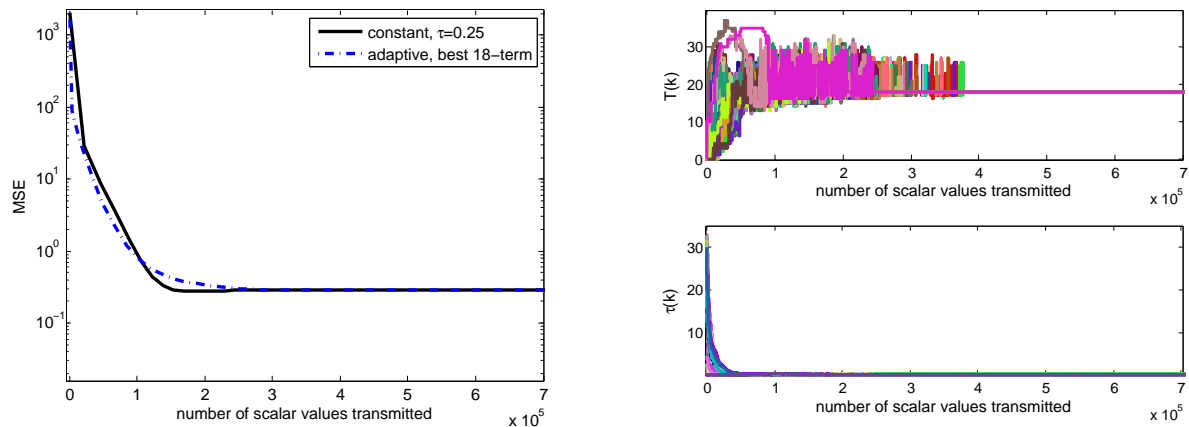


Fig. 14. Left panel: A comparison of selective gossip with centralized constant threshold value of  $\tau = 0.25$  and decentralized adaptive threshold mechanism with best  $m = 18$  terms. Averaged over 25 runs of the algorithm. Right panel: Evolution for one run of the algorithm. Top: Number of significant coefficients at every node, plotted on top of each other. Bottom: Threshold  $\tau$  over the iterations, at every node, plotted on top of each other.

in the threshold estimates.

We now compare the clairvoyant threshold mechanism (constant, preset  $\tau$ ) to the proposed adaptive mechanism through simulations. The simulation is performed for the same field as in Section V-A which is now sensed by 64 nodes over a  $8 \times 8$  grid. The linear transform that is used is the 2-level Haar wavelet transform. The simulation is run 25 times with different random seeds. Figure 14(a) illustrates the decrease in mean squared error as gossip iterations are performed for the centralized case with  $\tau = 0.25$  and for the decentralized adaptive case with  $m = 18$ . Note that for this signal, taking only the highest 18 coefficients is practically same as thresholding at  $\tau = 0.25$ . The constants are chosen as  $c_1 = 0.025$  and  $c_2 = 0.05$ .

The left panel of Figure 14 illustrates that the decentralized algorithm yields nearly the same MSE values as the clairvoyant algorithm with  $\tau = 0.25$ . The right panel of Figure 14 shows the behavior of the estimated values at nodes for number of significant coefficients and threshold estimates over time. All nodes converge to the required best 18-term approximation level, and even though the initial threshold values are very high and vary among nodes, every node is still able to converge to the desired threshold eventually.

## VII. DISCUSSION

In this paper we describe selective gossip, an algorithm for decentralized sparse approximation. Selective gossip is aimed at computing a vector of data, where some components of the vector are insignificant

and do not need to be computed exactly, but the locations of these components are not known in advance. Instead, selective gossip adaptively determines which components are significant while the computation is being carried out, and automatically adjusts where transmissions are invested in order to efficiently obtain a good approximation. We prove that the algorithm converges. We provide simulation results comparing selective gossip to parallel randomized gossip on elements of a vector and decentralized compression. We observe that selective gossip requires fewer scalar values to be transmitted while achieving same error value. Furthermore, we provide a decentralized adaptive threshold mechanism which removes the requirement for a fixed threshold. Selective gossip in conjunction with the adaptive threshold mechanism provides best  $m$ -term approximations of network data.

The future work includes investigating the rates of convergence for selective gossip. In an abstract level, the rates of convergence can be associated with the study of voter models from interacting particle systems [28]. In the voter model each node has binary value, i.e. vote. A node chooses a random neighbor with some probability and adopts the state of this neighbor. Hence the significance of component values in selective gossip can be seen as the votes in the voter model. For finite graphs, the authors of [29] show that the convergence time of voter model process is related to the hitting times of the random walks on the graph. Using analogy from this theory, the convergence of selective gossip can be related to hitting times.

The current version of selective gossip is implemented using standard, pair-wise randomized gossip as a building block. In order to further improve the rates of convergence we will investigate implementing selective gossip with the synchronous distributed averaging algorithm of [30]. This algorithm provides faster rates compared to randomized gossip at the cost of extra memory at nodes.

In this paper, we focused on compression of network data as an application of selective gossip. Our algorithm can also be used for distributed ranking in a mobile social network. An example application would be people having mobile devices and ranking songs, movies, etc. on their devices. In this case, selective gossip can be used to approximate highest ranked titles over the network in a decentralized fashion.

## REFERENCES

- [1] D. Üstebay, R. Castro, and M. Rabbat, "Selective gossip," in *Proc. The Third International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Aruba, Dutch Antilles, Dec. 2009.
- [2] J. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Massachusetts Institute of Technology, 1984.



- [3] R. Olfati-Saber, J. A. Fax, and R.M.Murray, “Consensus and cooperation in networked multi-agent systems,” in *Proc. IEEE Information Theory Workshop*, Porto, Portugal, May 2007.
- [4] A. Dimakis, S. Kar, J. Moura, M. Rabbat, and A. Scaglione, “Gossip algorithms for distributed signal processing,” *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.
- [5] M. Rabbat, J. Haupt, A. Singh, and R. Nowak, “Decentralized compression and predistribution via randomized gossiping,” in *Proc. Information Processing in Sensor Networks*, Nashville, TN, Apr. 2006.
- [6] M. Yildiz, F. Ciaranello, and A. Scaglione, “Distributed distance estimation for manifold learning and dimensionality reduction,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, Apr. 2009.
- [7] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Randomized gossip algorithms,” *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508–2530, June 2006.
- [8] A. Dimakis, A. Sarwate, and M. Wainwright, “Geographic gossip: Efficient aggregation for sensor networks,” in *Proc. Information Processing in Sensor Networks*, Nashville, TN, Apr. 2006.
- [9] F. Benezit, A. Dimakis, P. Thiran, and M. Vetterli, “Gossip along the way: Order-optimal consensus through randomized path averaging,” in *Proc. Allerton Conference on Communication, Control, and Computing*, Urbana-Champaign, IL, Sep. 2007.
- [10] T. Aysal, M. Yildiz, A. Sarwate, and A. Scaglione, “Broadcast gossip algorithms: Design and analysis for consensus,” in *Proc. IEEE Conference on Decision and Control*, Cancun, Mexico, Dec. 2008.
- [11] D. Üstebay, B. N. Oreshkin, M. J. Coates, and M. G. Rabbat, “Greedy gossip with eavesdropping,” *IEEE Transactions on Signal Processing*, vol. 58, no. 7, pp. 3765–3776, July 2010.
- [12] E. Yildiz and A. Scaglione, “Differential nested lattice encoding for consensus problems,” in *Proc. Information Processing in Sensor Networks*, Cambridge, MA, Apr. 2007.
- [13] W. Wang, M. Garofalakis, and K. Ramchandran, “Distributed sparse random projections for refinable approximation,” in *Proc. Information Processing in Sensor Networks*, Cambridge, MA, Apr. 2007.
- [14] G. Shen, S. Patten, and A. Ortega, “Energy-efficient graph-based wavelets for distributed coding in wireless sensor networks,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, Apr. 2009.
- [15] R. Wagner, R. Baraniuk, S. Du, D. Johnson, and A. Cohen, “An architecture for distributed wavelet analysis and processing in sensor networks,” in *Proc. Information Processing in Sensor Networks*, Nashville, TN, Apr. 2006.
- [16] F. Wuhib, M. Dam, and R. Stadler, “A gossiping protocol for detecting global threshold crossings,” *IEEE Transactions on Network and Service Management*, vol. 7, no. 1, pp. 42–57, Mar. 2010.
- [17] C. Castellano, S. Fortunato, and V. Loreto, “Statistical physics of social dynamics,” *Rev. Mod. Phys.*, vol. 81, no. 2, pp. 591–646, May 2009.
- [18] V. Blondel, J. Hendrickx, and J. Tsitsiklis, “On Krause’s multi-agent consensus model with state-dependent connectivity,” *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2586–2597, Nov. 2009.
- [19] G. Weisbuch, G. Deffuant, F. Amblard, and J.-P. Nadal, “Meet, discuss and segregate!” *Complexity*, vol. 7, no. 3, pp. 55–63, 2002.
- [20] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Belmont, MA: Athena Scientific, 1997.

- [21] M. Vetterli, “Wavelets, approximation, and compression,” *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 59–73, Sep. 2001.
- [22] R. DeVore, “Nonlinear approximation,” *Acta numerica*, vol. 7, pp. 51–150, 1998.
- [23] R. R. Coifman and M. Maggioni, “Diffusion wavelets,” *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 53 – 94, 2006.
- [24] F. Chung, *Spectral Graph Theory*. CBMS-AMS, 1997, no. 92.
- [25] D. Kempe and F. McSherry, “A decentralized algorithm for spectral analysis,” in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, Chicago, IL, Jun. 2004.
- [26] “Intel Lab data,” June 2004. [Online]. Available: <http://db.csail.mit.edu/labdata/labdata.html>
- [27] “CIMIS data,” 2010. [Online]. Available: <http://www.cimis.water.ca.gov>
- [28] T. M. Liggett, *Interacting Particle Systems*. Springer-Verlag, 1985.
- [29] D. Aldous and J. A. Fill, “Reversible markov chains and random walks on graphs,” manuscript, 1994. [Online]. Available: <http://www.stat.berkeley.edu/~aldous/RWG/book.html>
- [30] B. Oreshkin, M. Coates, and M. Rabbat, “Optimization and analysis of distributed averaging with short node memory,” *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2850 –2865, May 2010.