On Quantized Consensus by Means of
Gossip Algorithm – Part I: Convergence Proof

Javad Lavaei and Richard M. Murray

Abstract— This paper is concerned with the distributed averaging problem subject to a quantization constraint. Given a group of agents associated with scalar numbers, it is assumed that each pair of agents can communicate with a prescribed probability, and that the data being exchanged between them is quantized. In this part of the paper, it is proved that the stochastic gossip algorithm proposed in a recent paper leads to reaching the quantized consensus. Some important steady-state properties of the system (after reaching the consensus) are also derived. The results developed here hold true for any arbitrary quantization, provided that the tuning parameter of the gossip algorithm is chosen properly. The expected value of the convergence time is lower and upper bounded in the second part of the paper.

I. INTRODUCTION

Consider a group of agents, each of which is associated with some data such as a real number or an image. The problem of contriving a strategy by means of which all agents can update themselves so that they ultimately agree upon some universal shared data is called the *consensus* or *state agreement* problem [1], [2]. Consensus has a long history in computer science, particularly in distributed computation where a program is divided into parts that run simultaneously on multiple computers communicating over a network [3], [4].

There are many important real-world problems whose treatment is contingent upon the notion of consensus. In the load-balancing problem, the tasks of disparate processors are to be equalized in order to refrain from overloading the processors [5], [6]. In the synchronization of coupled oscillators arising in systems biology, the oscillation frequencies of all agents are desired to become equal [7], [8]. In multi-agent coordination and flocking, there are a number of applications in which the state-agreement problem appears [9], [10]. For instance, the heading angles of different mobile agents may be required to be aligned [11]. In a sensor network comprising a set of sensors measuring the same quantity in a noisy environment, the state estimates of different agents must be averaged [12]. A more complete survey on these topics is given in the recent paper [2].

Consider the distributed average consensus in which the values owned by the agents are to be averaged in a distributed fashion. Since it may turn out in some applications that all agents cannot update their numbers synchronously, the gossip algorithm has been widely exploited by researchers to handle

the averaging problem asynchronously [13], [14]. This type of algorithm selects a pair of agents at each time instance, and updates their values based on some averaging policy. The consensus problem in the context of gossip algorithm has been thoroughly investigated in the literature [15], [16], [17], [18].

In light of communication constraints, the data being exchanged between each pair of agents is normally quantized. This has given rise to the emergence of quantized gossip algorithms. The notion of quantized consensus is defined in [17] for the case when quantized values (integers) are to be averaged over a connected network with digital communication channels. That paper shows that the quantized gossip algorithm leads to reaching the quantized consensus. This result is extended in [18] to the case when the quantization is uniform, and the initial numbers owned by the agents are reals (as opposed to being integers). The paper [18] shows that the quantized gossip algorithm works for a particular choice of the updating parameter, although it conjunctures that this result is valid for a wide range of updating parameters. A related paper on quantized consensus gives a synchronous algorithm in order to reach a consensus with arbitrary precision, at the cost of not preserving the average of the initial numbers [19].

Part I of the current work starts with proving the above-mentioned property of quantized consensus. More precisely, a weighted connected graph is considered together with a set of scalars sitting on its vertices. The weight of each edge represents the probability of establishing a communication between its corresponding vertices through the updating procedure. It is shown that the quantized consensus is reached under the stochastic gossip algorithm proposed in [18], for a broad range of updating parameters. This result holds true for any arbitrary quantizer, including uniform and logarithmic ones. Some elegant properties of the system in the steady state (after reaching the consensus) are subsequently derived. The second part of the paper deals with the expected value of the time at which the consensus is reached [20]. This quantity (in the worst case) is lower and upper bounded in terms of the Laplacian of the weighted graph. A convex optimization is then proposed to investigate what set of weighes on the edges results in a small convergence time.

This paper is organized as follows. Some preliminaries are presented in Section II, and the problem is formulated accordingly. The convergence proof is provided in Section III for uniform quantizers, and is generalized to arbitrary quantizers in Section IV. The results are illustrated in Section V through simulations. Some concluding remarks are finally

drawn in Section VI.

II. PROBLEM FORMULATION

Consider a connected weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{P})$, where:

- $\mathcal{V} := \{v_1, v_2, ..., v_\nu\}$ is the set of vertices of $\mathcal{G}$;
- $\mathcal{E}$ is the set of undirected edges of $\mathcal{G}$;
- $\mathcal{P} := \{p_{ij}\}_{i,j}$ is the set of weights assigned to the edges of $\mathcal{G}$.

Assume that:

- The quantity $\sum p_{ij}$ is equal to 1, where the sum is taken over all numbers $i, j \in \nu := \{1, 2, ..., \nu\}$ such that $i \leq j$.
- The number p_{ij} $(i, j \in \nu)$ is equal to zero if $(i, j) \notin \mathcal{E}$; otherwise, it is strictly positive. In particular, $p_{11}, p_{22}, ..., p_{\nu\nu}$ are all equal to zero.

The scalar p_{ij} associated with the edge (i, j) represents the probability of choosing the edge (i, j) when an edge of $\mathcal{G}$ is to be picked at random. Suppose that a real number x_i has been assigned to the vertex v_i, for all $i \in \nu$. Let $q(x) : \mathbf{R} \to \mathbf{R}$ be a given quantization operator, which can be, for instance, a logarithmic or constant quantizer. In what follows, a quantized gossip algorithm is presented [18].

Stochastic Gossip (SG) Algorithm:
Step 1: Given a positive real ε, set $k = 0$. Define $x_i[0] := x_i$, for all $i \in \nu$.
Step 2: Pick an edge of $\mathcal{G}$ at random.
Step 3: Suppose that the ending vertices of the edge selected in step 2 possess the values $x_i[k]$ and $x_j[k]$. Perform the following updates:

$$
\begin{aligned}
x_i[k+1] &= x_i[k] + \varepsilon \times \big(q(x_j[k]) - q(x_i[k])\big), \\
x_j[k+1] &= x_j[k] + \varepsilon \times \big(q(x_i[k]) - q(x_j[k])\big), \quad (1) \\
x_q[k+1] &= x_q[k], \quad \forall q \in \nu \backslash \{i, j\}
\end{aligned}
$$

Step 4: Increase k by 1 and jump to step 2.

Throughout this paper, the symbol $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ refers to the weighted graph $\mathcal{G}$, whereas the symbol $\mathcal{G}(\mathcal{V}, \mathcal{E})$ refers to the graph $\mathcal{G}$ with the weights on its edges removed. For simplicity, the short-hand notation:

$$
\mathbf{x}[k] := \begin{bmatrix} x_1[k] & x_2[k] & \cdots & x_\nu[k] \end{bmatrix}, \quad k \in \mathbf{N} \cup \{0\} \quad (2)
$$

will be used henceforth. The next definition is given in [18] for the case when $q(\cdot)$ is a uniform quantizer rounding each number to its nearest integer.

Definition 1: Given a quantization-based protocol $\mathcal{C}$ acting on $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (e.g. the deterministic gossip algorithm), assume that the vector $\mathbf{x}[k]$ denotes the values on the vertices of $\mathcal{G}$ at time k, obtained using this protocol. It is said that the *quantized consensus* is reached for the graph $\mathcal{G}$ under the protocol $\mathcal{C}$ if for every arbitrary initial state $\mathbf{x}[0] \in \mathbf{R}^\nu$, there exists a natural number k_0 such that:

$$
|x_i[k] - x_{\text{ave}}| < 1, \quad \forall k \geq k_0, \ \forall i \in \nu \quad (3)
$$

where $x_{\text{ave}} := \frac{x_1[0] + x_2[0] + \cdots + x_\nu[0]}{\nu}$.

In line with the above definition, if the protocol $\mathcal{C}$ is stochastic (e.g. the SG algorithm), one would say that the quantized consensus is reached *almost surely* if there exists a number $k_0 \in \mathbf{N}$, with probability 1, for which inequality (3) holds. In the rest of the paper, the short name *consensus* will be used for *quantized consensus*.

It is shown in [18] that if the quantizer $q(x)$ is uniform, the consensus is reached almost surely for the graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ under the SG algorithm, provided $\varepsilon = 0.5$. That paper also conjectures that the same result holds true for every positive number $\varepsilon < 0.5$, while it may not be true for $\varepsilon > 0.5$ (as simulation confirms). The primary objective of the present work is to prove this property, not only for a uniform quantizer but also for any arbitrary type of quantizer.

III. MAIN RESULTS

In the remainder of the paper, assume that $\varepsilon \in (0, 0.5]$ (unless otherwise stated). Let x_{max} and x_{min} be defined as:

$$
x_{max} := \max_{i \in \nu} \lceil x_i \rceil, \quad x_{min} := \min_{i \in \nu} \lfloor x_i \rfloor \quad (4)
$$

where $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ stand for the ceiling and floor operators, respectively.

Definition 2: Define $\mathcal{S}$ to be the set of all ν-tuple $(\alpha_1, \alpha_2, ..., \alpha_\nu)$ such that $\alpha_i \in [x_{min}, x_{max}]$ and, in addition, $\alpha_i - x_i$ is an integer multiple of ε, for every $i \in \nu$.

Definition 3: Define the following quantities:

$$
\begin{aligned}
\eta_1 &:= \max \left\{ \frac{2k+1}{2} \bigg| k \in \mathbf{Z}, \ \frac{2k+1}{2} \leq x_{ave} \right\}, \\
\eta_2 &:= \min \left\{ \frac{2k+1}{2} \bigg| k \in \mathbf{Z}, \ \frac{2k+1}{2} \geq x_{ave} \right\}
\end{aligned} \quad (5)
$$

where $\mathbf{Z}$ denotes the set of integers.

Definition 4: Let $\mathcal{S}_o$ and $\mathcal{S}_o(\mu)$, $\mu \in \mathbf{R}$, be defined as follows:

$$
\mathcal{S}_o := \big\{ (\alpha_1, \alpha_2, ..., \alpha_\nu) \in \mathcal{S} \big| \ \alpha_i \in (\eta_1, \eta_2], \ \forall i \in \nu \big\}
$$

and:

$$
\mathcal{S}_o(\mu) := \big\{ (\alpha_1, ..., \alpha_\nu) \in \mathcal{S} \big| \ \alpha_i \in (\mu - \varepsilon, \mu + \varepsilon], \ \forall i \in \nu \big\}
$$

Definition 5: Define the distance function $d_\varepsilon(\cdot, \mathcal{S}_o) : \mathcal{S} \to \mathbf{Z}$ as:

$$
d_\varepsilon(\boldsymbol{\alpha}, \mathcal{S}_o) := \min_{\boldsymbol{\beta} \in \mathcal{S}_o} \frac{|\boldsymbol{\alpha} - \boldsymbol{\beta}|_1}{\varepsilon}, \quad \forall \boldsymbol{\alpha} \in \mathcal{S} \quad (6)
$$

where $|\cdot|_1$ denotes the L_1 norm. In the same way, define $d_\varepsilon(\boldsymbol{\alpha}, \mathcal{S}_o(\mu))$ for every real μ (note that $d_\varepsilon(\boldsymbol{\alpha}, \mathcal{S}_o(\mu))$ is equal to zero if $\boldsymbol{\alpha} \in \mathcal{S}_o(\mu)$).

Throughout the rest of this section, assume that $q(x)$ is a uniform quantizer, i.e., it rounds each real number x to its nearest integer (by convention, assume that $q(r + 0.5) = r$, for all integers r). The results will be later extended to a general quantizer $q(x)$.

A. Geometric intuition

To better understand the arguments made in the present work, it is desired to illustrate the above definitions graphically. Notice that although the graph $\mathcal{G}$ is coordinate free, when it comes to assigning real numbers to its vertices, it is beneficial to incorporate the topology of the graph and its corresponding initial state into a new graph whose geometry matters. Hence, perform the following operations on the x-y plane:

- Draw the horizontal lines $y = x_{\min}$, $y = x_{\max}$, $y = x_{\text{ave}}$, $y = \eta_1$ and $y = \eta_2$.
- For every $i \in \boldsymbol{\nu}$, mark all points (i, p) lying between the lines $y = x_{\min}$ and $y = x_{\max}$ for which $p - x_i(0)$ is an integer multiple of ε.
- Draw the graph $\mathcal{G}$ in this plane in such a way that its vertex v_i is placed in the coordinates $(i, x_i(0))$, for all $i \in \boldsymbol{\nu}$.

Denote the coordinated graph drawn in this manipulated plane with $\vec{\mathcal{G}}$. As an example, if $\mathcal{G}$ contains 4 vertices with the edges $\{(1,2),(2,3),(3,4),(4,1)\}$ and the initial state $\mathbf{x}[0] = (3.875, 1.625, 1.625, 3.750)$, then the corresponding coordinated graph $\vec{\mathcal{G}}$ will turn out to be the one depicted in Figure 1. When the SG algorithm is run on the graph
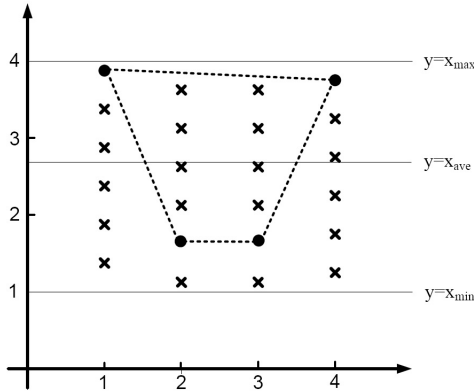

Fig. 1. An example of the coordinated graph $\vec{\mathcal{G}}$ with four vertices (the lines $y = \eta_1$ and $y = \eta_2$ have not been drawn in this figure in order to keep it simple).

$\mathcal{G}$, the values sitting on the vertices change. This makes the graph $\vec{\mathcal{G}}$ move in the plane (each vertex moves up or down vertically). Define $\vec{\mathcal{G}}(k)$ to be the corresponding coordinated graph at time k ($k \in \mathbf{N}$). It can be concluded from (1) that the vertices of $\vec{\mathcal{G}}(k)$ are always located on the marked points of the plane. For simplicity, assume that there exists no marked point on the horizontal lines drawn above. The graphical interpretation of Definitions 2 to 5 is as follows:

- Each element of $\mathcal{S}$ corresponds to a collection of ν marked points that lie in ν distinct columns (a column refers to one of the vertical lines $x = i$, $i = 1, 2, ..., \nu$).
- Each element of $\mathcal{S}_o$ corresponds to a collection of ν marked points lying in ν distinct columns which are confined by the lines $y = \eta_1$ and $y = \eta_2$. The interpretation of the set $\mathcal{S}_o(\mu)$ is similar.

- Let $\boldsymbol{\alpha}$ denote the set of values sitting on the vertices of the graph $\mathcal{G}$ at some time instance $k \in \mathbf{N}$. For every vertex of the coordinated graph $\vec{\mathcal{G}}(k)$, count the number of marked points lying (vertically) between that vertex and the line $y = \mu$, and add up these values for all vertices. The resulting number is nothing but the function $d_\varepsilon(\boldsymbol{\alpha}, \mathcal{S}_o(\mu))$. As a result, this function is intended to measure the distance between the vertices of the coordinated graph $\vec{\mathcal{G}}(k)$ and the line $y = \mu$.

Several results will be provided in this section, whose corresponding geometrical interpretations are listed below:

- If the graph $\vec{\mathcal{G}}(k)$ belongs to the set $\mathcal{S}_o$ for some $k \in \mathbf{N}$, then it can never be moved under the operations of the SG algorithm (Lemma 1).
- Given integers $r \in [x_{\min}, x_{\max}]$ and $k \in \mathbf{N}$, if the graph $\vec{\mathcal{G}}(k)$ belongs to the set $\mathcal{S}_o(r + 0.5)$, then its movement under the SG algorithm will be confined within this set (Lemma 1).
- Given an integer $r \in [x_{\min}, x_{\max}]$, the function $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(r+0.5))$ is non-increasing with respect to k, i.e., the coordinated graph $\vec{\mathcal{G}}(k)$ always moves towards the line $y = r + 0.5$ (in the sense defined earlier) as k increases (Lemma 2).
- Given an integer $r \in [x_{\min}, x_{\max}]$, there exists a finite time $k \in \mathbf{N}$ for which the coordinated graph $\vec{\mathcal{G}}(k)$ either lies on one side of the line $y = r + 0.5$ or belongs to the invariant set $\mathcal{S}_o(r + 0.5)$ (Lemma 3).
- There exists a finite time k_1 such that the coordinated graph $\vec{\mathcal{G}}(k_1)$ is placed entirely in one of the invariant sets $\mathcal{S}_o(\eta_1)$, $\mathcal{S}_o(\eta_2)$ and $\mathcal{S}_o$ (Theorem 2).

The last property given above is indeed the main result of this section. Notice that for the example illustrated in Figure 1, the fact that $\vec{\mathcal{G}}(k)$ belongs to $\mathcal{S}_0$, $\mathcal{S}_o(\eta_1)$ or $\mathcal{S}_o(\eta_2)$ for large enough k's implies that the vertices of this graph eventually lie in the set of encircled points shown in Figures 2a, 2b or 2c. This result characterizes the steady-state behavior of the graph system under the SG algorithm.

B. Mathematical proofs

The results stated in the preceding subsection will be rigorously proved in the sequel. Observe that the SG algorithm is stochastic in the sense that an edge must be chosen *at random* at each time update. The deterministic version of this algorithm, referred to as the deterministic gossip (DG) algorithm, can be obtained by replacing its step 2 with the following:

Step 2: Pick an edge of $\mathcal{G}$ <u>arbitrarily</u> (at the discretion of the user).

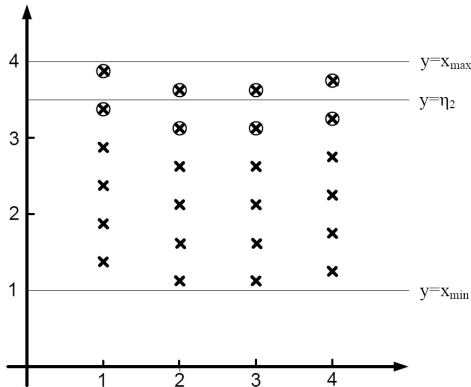The next theorem relates the convergence of the DG algorithm to that of the SG algorithm.

Theorem 1: Assume that step 2 of the DG algorithm (i.e. picking an edge arbitrarily) can be taken in such a way that the consensus is reached for the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ under this algorithm. Then, the consensus is reached almost surely for the graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ under the SG algorithm .
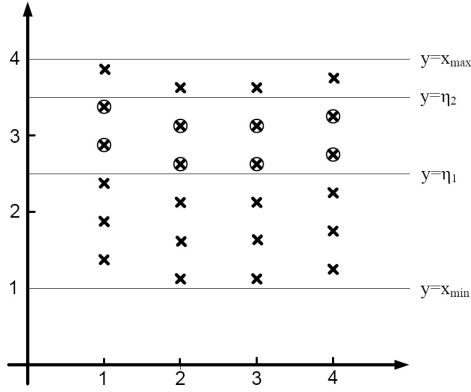
(a)
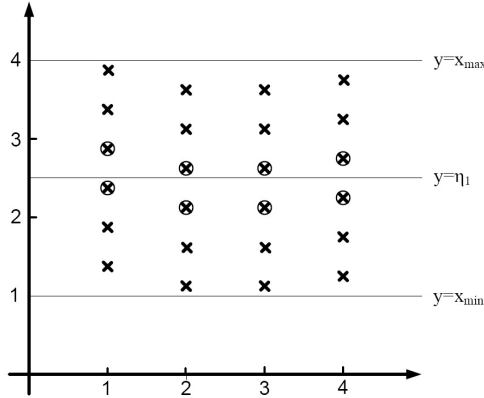
(b)

(c)

Fig. 2. The encircled points in these figures characterize the elements of the sets $\mathcal{S}_0$, $\mathcal{S}_o(\eta_1)$ and $\mathcal{S}_o(\eta_2)$, respectively.

Proof: Apply the SG algorithm to the graph $\mathcal{G}$ with the initial state $\mathbf{x}[0]$. Using induction, one can easily conclude from equation (1) that:

i) $x_i[k]$ is always in the interval $[x_{min}, x_{max}]$, for all $k \in \mathbf{N} \cup \{0\}$ and $i \in \boldsymbol{\nu}$.

ii) The vector $\mathbf{x}[k]$ belongs to the set $\mathcal{S}$, for all $k \in \mathbf{N} \cup \{0\}$.

This signifies that the state of the graph system can take values only in the finite set $\mathcal{S}$. Now, form a new graph $\mathcal{R}$ as follows:

- Put $|\mathcal{S}|$ vertices corresponding to the elements of the set $\mathcal{S}$.
- Draw two directed edges between every pair of vertices (the edges must have opposite orientations).
- Assign a number to each directed edge $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, where $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathcal{S}$, to represent the probability of transition from $\boldsymbol{\alpha}$ to $\boldsymbol{\beta}$ via the SG algorithm in only one iteration.

It is easy to verify that if the SG algorithm is run on the graph $\mathcal{G}$ with the initial state $\mathbf{x}[0]$, it will generate a random walk on the graph $\mathcal{R}$ starting from the vertex $\mathbf{x}[0]$. Let $\mathcal{R}_0$ denote the subgraph of $\mathcal{R}$ that corresponds to the subset of $\mathcal{S}$ determining the quantized consensus. The assumption of the theorem can be interpreted as *there is a walk from every vertex of $\mathcal{R}$ to the subgraph $\mathcal{R}_0$.* It is follows from a well-known theorem in the Markov chain theory that every infinite random walk almost surely ends up in this subset of vertices. This completes the proof. ∎

Theorem 1 states that in order to prove the convergence of the stochastic gossip algorithm, it suffices to show that of its deterministic version. Hence, this converse statement will be proved in the sequel.

Lemma 1: Apply the SG algorithm to the graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ with the initial state $\mathbf{x}[0]$.

- Suppose that $\mathbf{x}[k]$ belongs to the set $\mathcal{S}_o$ for some non-negative integer k. The equality $\mathbf{x}[k+1] = \mathbf{x}[k]$ holds. In other words, each element of $\mathcal{S}_o$ is an equilibrium point of the discrete-time system.
- Assume that $\mathbf{x}[k]$ belongs to the set $\mathcal{S}_o(r + 0.5)$, for some integers k and r. The state $\mathbf{x}[k+1]$ is in the set $\mathcal{S}_o(r+0.5)$ as well. In other words, this set is invariant under the underlying algorithm.

Proof: The proof is straightforward, and is omitted for brevity. ∎

Lemma 2: Apply the SG algorithm to the graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ with the initial state $\mathbf{x}[0]$. Given $r \in \mathbf{Z}$, the following inequality holds for every nonnegative integer k:

$$d_\varepsilon(\mathbf{x}[k+1], \mathcal{S}_o(r + 0.5)) \leq d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(r + 0.5)) \quad (7)$$

Proof: Assume that the edge (i, j) is chosen at the $(k+1)^{\text{th}}$ time update, and that $x_i[k] \leq x_j[k]$. There are a number of possibilities as follows:

- $x_i[k] - r - 0.5 > 0$ *and* $x_j[k] - r - 0.5 > 0$: It can be easily shown that:

$$x_i[k], x_j[k], x_i[k+1], x_j[k+1] > r + 0.5 \quad (8)$$

The above inequalities together with the equality:

$$x_i[k] + x_j[k] = x_i[k+1] + x_j[k+1] \quad (9)$$

allow us to conclude that:

$$d_\varepsilon(\mathbf{x}[k+1], \mathcal{S}_o(r+0.5)) = d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(r+0.5)) \quad (10)$$

- $x_i[k] - r - 0.5 \le 0$ *and* $x_j[k] - r - 0.5 \le 0$*:* It is easy to observe that:

$$x_i[k], x_j[k], x_i[k+1], x_j[k+1] \le r + 0.5 \quad (11)$$

This leads to the equality (10) (as before).

- $x_i[k] - r - 0.5 \le$ *and* $x_j[k] - r - 0.5 > 0$*:* Similar to the previous cases, it can be shown that inequality (7) holds. ∎

Lemma 3: Given $r \in \mathbf{Z}$, apply the DG algorithm to the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with the initial state $\mathbf{x}[0]$. At each time update $k \in \mathbf{N}$, select an edge of the graph (in step 2 of the algorithm) such that the function $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(r + 0.5))$ is minimized. There exists a natural number k_0 for which either of the following cases occurs:

i) $\mathbf{x}[k]$ is in the invariant set $\mathcal{S}_o(r + 0.5)$, for all $k \ge k_0$.
ii) $x_1[k] - r - 0.5, x_2[k] - r - 0.5, ..., x_\nu[k] - r - 0.5$ are either all *negative* or all *strictly positive*, for every $k \ge k_0$.

Proof: Since $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(r+0.5))$ is a nonnegative integer-valued decreasing function (by Lemma 2), there exists a number k_0 with the property:

$$d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(r+0.5)) = d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r+0.5)), \quad \forall k \ge k_0 \quad (12)$$

If $d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5)) = 0$, then case (i) explained in the statement of the lemma definitely occurs. It is desired to prove that if $d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5)) \ne 0$, then case (ii) takes place. To this end, notice that if $x_i[k] - r - 0.5, \forall i \in \nu$, are negative (strictly positive) for some proper k, so are $x_i[k + 1] - r - 0.5, \forall i \in \nu$. This implies that it suffices to prove case (ii) only for $k = k_0$.

To prove by contradiction, assume that there exist two integers $i, j \in \nu$ such that:

$$x_i[k_0] > r + 0.5, \quad x_j[k_0] \le r + 0.5 \quad (13)$$

Since the graph $\mathcal{G}$ is connected, the above two inequalities yield that there are two integers $\mu_1, \mu_2 \in \nu$ subject to:

- (μ_1, μ_2) is an edge of the graph $\mathcal{G}$.
- $x_{\mu_1}[k_0] > r + 0.5$ and $x_{\mu_2}[k_0] \le r + 0.5$.

If $x_{\mu_1}[k_0] > r + 0.5 + \varepsilon$ or $x_{\mu_2}[k_0] \le r + 0.5 - \varepsilon$, then following the proof of Lemma 2, one can conclude that choosing the edge (μ_1, μ_2) at time $k_0 + 1$ in step 2 of the DG algorithm results in the reduction of $d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5))$, i.e.:

$$d_\varepsilon(\mathbf{x}[k_0 + 1], \mathcal{S}_o(r + 0.5)) < d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5)) \quad (14)$$

which is impossible in light of equality (12). Thus:

$$r + 0.5 < x_{\mu_1}[k] \le r + 0.5 + \varepsilon, \quad (15a)$$
$$r + 0.5 \ge x_{\mu_2}[k] > r + 0.5 - \varepsilon \quad (15b)$$

Consider an arbitrary vertex connected to v_{μ_2}, and denote it with v_{μ_3} (if such a vertex does not exist, find a vertex connected to v_{μ_1} instead). It is desired to prove that:

$$r + 0.5 - \varepsilon < x_{\mu_3}[k_0] \le r + 0.5 + \varepsilon \quad (16)$$

To this end, consider the following scenarios:

- $x_{\mu_3}[k_0]$ *is greater than* $r + 0.5$: It results from (15b) that if the inequality $x_{\mu_3}[k_0] \le r + 0.5 + \varepsilon$ does not hold, then choosing the edge (μ_2, μ_3) at time $k_0 + 1$ through the DG algorithm will reduce the storage function $d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5))$, which is impossible by (12).
- $x_{\mu_3}[k_0]$ *is less than or equal to* $r + 0.5$: If the relation $r + 0.5 - \varepsilon < x_{\mu_3}[k_0]$ does not hold, run step 2 of the DG algorithm at times $k_0 + 1$ and $k_0 + 2$ as follows:
  - At time $k_0 + 1$, choose the edge (μ_1, μ_2) which gives the updates (in light of (15)):

$$x_{\mu_1}[k_0 + 1] = x_{\mu_1}[k_0] - \varepsilon,$$
$$x_{\mu_2}[k_0 + 1] = x_{\mu_2}[k_0] + \varepsilon \quad (17)$$

    Therefore:

$$d_\varepsilon(\mathbf{x}[k_0 + 1], \mathcal{S}_o(r + 0.5)) = d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5)) \quad (18)$$

  - At time $k_0 + 2$, choose the edge (μ_2, μ_3). Equation (17) leads to:

$$r + 0.5 < x_{\mu_2}[k_0 + 1] \le r + 0.5 + \varepsilon,$$
$$x_{\mu_3}[k_0 + 1] = x_{\mu_3}[k_0] \le r + 0.5 - \varepsilon \quad (19)$$

    Thus, one can show that:

$$d_\varepsilon(\mathbf{x}[k_0 + 2], \mathcal{S}_o(r + 0.5)) \le$$
$$d_\varepsilon(\mathbf{x}[k_0 + 1], \mathcal{S}_o(r + 0.5)) - 1 \quad (20)$$

    which is impossible by equation (12).

This concludes the validity of inequality (16). Since the graph is connected, there is a path from v_{μ_2} to any other vertex in $\mathcal{V}$. One can continue the argument made above (on the vertex v_{μ_3}) for the vertices of such paths successively to deduce:

$$r + 0.5 - \varepsilon < x_i[k_0] \le r + 0.5 + \varepsilon, \quad \forall i \in \nu \quad (21)$$

The above inequality signifies that $d_\varepsilon(\mathbf{x}[k_0], \mathcal{S}_o(r + 0.5))$ is equal to zero, while this quantity was earlier assumed to be nonzero. This contradiction completes the proof. ∎

Theorem 2: Apply the DG algorithm to the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with the initial state $\mathbf{x}[0]$. Step 2 of this algorithm (i.e. selecting an edge arbitrarily) can be taken appropriately so that there exists a positive number k_1 for which one of the following cases takes place:

i) $\mathbf{x}[k]$ belongs to the set $\mathcal{S}_o$, for all $k \ge k_1$.
ii) $\mathbf{x}[k]$ belongs to the set $\mathcal{S}_o(\eta_1)$, for all $k \ge k_1$.
iii) $\mathbf{x}[k]$ belongs to the set $\mathcal{S}_o(\eta_2)$, for all $k \ge k_1$.

Proof: Define the storage functions:

$$V_1[k] := d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(\eta_1)),$$
$$V_2[k] := d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(\eta_2)) \quad (22)$$

In the course of taking step 2 of the DG algorithm, select an edge at each time update $k \in \mathbf{N}$ such that the function $V_1[k]$ is minimized (as explained in the statement of Lemma 3). Halt at a time k_0, where $V_1[k]$ reaches its minimum and remains constant. By the preceding lemma, one of the following cases happens:

- **x**$[k]$ *is in the invariant set* $\mathcal{S}_o(\eta_1)$, *for all* $k \geq k_0$: If this is the case, the proof is complete.
- $x_1[k] - \eta_1, x_2[k] - \eta_1, ..., x_\nu[k] - \eta_1$ *are all negative, for every* $k \geq k_0$: Since x_{ave} is greater than or equal to η_1, and is also identical to the average of the numbers $x_1[k], ..., x_\nu[k]$, this case is ruled out, unless $x_1[k] = x_2[k] = \cdots = x_\nu[k] = x_{ave} = \eta$. Nevertheless, this implies that **x**$[k] \in \mathcal{S}_o(\eta_1)$.
- $x_1[k] - \eta_1, x_2[k] - \eta_1, ..., x_\nu[k] - \eta_1$ *are all strictly positive, for every* $k \geq k_0$: At time $k = k_0$, ignore the mission of minimizing $V_1[k]$, and after this time take step 2 of the DG algorithm so that the Lyapunov function $V_2[k]$ is minimized at each time update. Notice that since all entries of **x**$[k_0]$ are greater than η_1, they can never go beyond this limit at a future time. Using Lemma 3, it can be argued that there exists a natural number $k_1 > k_0$ for which one of the following cases occurs:
  - **x**$[k]$ *is in the invariant set* $\mathcal{S}_o(\eta_2)$, *for all* $k \geq k_0$: If this is the case, the proof is complete.
  - $x_1[k] - \eta_2, x_2[k] - \eta_2, ..., x_\nu[k] - \eta_2$ *are all negative, for every* $k \geq k_0$: It follows from this case that **x**$[k]$ belongs to the set $\mathcal{S}_o$, for all $k \geq k_1$.
  - $x_1[k] - \eta_2, x_2[k] - \eta_2, ..., x_\nu[k] - \eta_2$ *are all strictly positive, for every* $k \geq k_0$: This case can be simply ruled out, by adopting an argument similar to the one made above. ∎

Theorems 1 and 2 give rise to the conclusion that the consensus is reached almost surely for the graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ under the SG algorithm, for all $\varepsilon \in (0, 0.5]$. Given a set $\mathcal{M} \subset \mathbf{R}^\nu$, define the diameter of $\mathcal{M}$ to be the supremum of the Euclidean distance between every two points in $\mathcal{M}$.

Remark 1: Definition 1 states that if the consensus is reached at time k_0, the state **x**$[k]$ belongs to the box $[x_{ave} - 1, x_{ave} + 1]^\nu$, for every $i \in \boldsymbol{\nu}$ and $k \geq k_0$. In contrast, it can be deduced from Theorems 1 and 2 that there exists a positive integer $k_1 \geq k_0$ such that **x**$[k]$, $\forall k \geq k_1$, belongs to one of the sets $\mathcal{S}_o$, $\mathcal{S}_o(\eta_1)$, or $\mathcal{S}_o(\eta_2)$. In this regard, two points can be made as follows:

- The diameter of the set given by Definition 1 is equal to 2, whereas that of each of the sets $\mathcal{S}_o$, $\mathcal{S}_o(\eta_1)$ and $\mathcal{S}_o(\eta_2)$ is at most 1. This implies that a more precise definition of consensus can be provided in terms of these sets.
- If **x**$[k]$ in the steady state (for large enough k's) is not constant (almost surely) and can oscillate, it should then belong to either $\mathcal{S}_o(\eta_1)$ or $\mathcal{S}_o(\eta_2)$, which are both of diameter 2ε (see Lemma 1). Note that the diameter of these sets can become arbitrarily small by rendering an appropriate ε. This implies that running the gossip algorithm for a small ε either makes the steady state constant or permits it to oscillate in a set with a small diameter (2ε). In the latter case, $x_i[k]$ can oscillate between only two numbers of difference ε (due to the definition of $\mathcal{S}_o(\mu)$, $\mu \in \mathbf{R}$).

To clarify Remark 1, consider the nominal values $x_{ave} =$ 10.6 and $\varepsilon = 0.2$. The definition of consensus borrowed from [18] states that there exists a positive integer k_0 such that:

$$9.6 < x_1[k], ..., x_\nu[k] < 11.6, \quad \forall k \geq k_0 \qquad (23)$$

In contrast, Theorem 2 asserts that there exists a number k_1 so that:

$$10.3 < x_1[k], ..., x_\nu[k] \leq 10.7, \quad \forall k \geq k_1 \qquad (24)$$

or:

$$10.5 < x_1[k], ..., x_\nu[k] \leq 11.5, \quad \forall k \geq k_1 \qquad (25)$$

(note that case (iii) in Theorem 2 is ruled out in this example, as the average of the entries of **x**$[k]$ cannot be smaller than all entries of **x**$[k]$). Comparing (23) with (24) and (25), one can simply observe that a more precise description of the steady state values on the vertices of $\mathcal{G}$ is delineated by (24) and (25). Besides, notice that if $9.6 < x_1[k], ..., x_\nu[k] < 11.6$ for some integer k, it may not be true that $9.6 < x_1[k+1], ..., x_\nu[k+1] < 11.6$ (because this region does not correspond to an invariant set in general, whereas $\mathcal{S}_o(10.5)$ and $\mathcal{S}_o$ are both invariant).

IV. GENERALIZATION TO ARBITRARY QUANTIZERS

Let $q(x) : \mathbf{R} \to \mathbf{R}$ be a general quantization operator characterized as follows:

$$q(x) = \begin{cases} L_i & \text{if} \quad x \in [L_i, \bar{L}_i] \\ L_{i+1} & \text{if} \quad x \in (\bar{L}_i, L_{i+1}] \end{cases} \qquad \forall i \in \mathbf{Z} \qquad (26)$$

where $\{L_i\}_{-\infty}^\infty$ is a monotonically increasing sequence of integers representing the quantization levels, and:

$$\bar{L}_i := \frac{L_i + L_{i+1}}{2}, \quad \forall i \in \mathbf{Z} \qquad (27)$$

The scalar quantities L_i and $\bar{L}_i$ will be referred to as *level* and *splitting level*, respectively. The convergence proof developed in Section III can be readily extended, provided Definitions 1, 3 and 4 are expressed in the general case. This is carried out in the following.

Revised Definition 1: Given a quantization-based protocol $\mathcal{C}$ acting on $\mathcal{G}(\mathcal{V}, \mathcal{E})$, denote with **x**$[k]$, $k \in \mathbf{N} \cup \{0\}$, the vector of values sitting on the vertices of $\mathcal{G}$ at time k, obtained using this protocol. It is said that the (quantized) consensus is reached for the graph $\mathcal{G}$ under the protocol $\mathcal{C}$ if for every arbitrary initial state **x**$[0] \in \mathbf{R}^\nu$, there exist a natural number k_0 and an integer μ such that either of the following sets of relations holds:

$$\begin{cases} \sum_{i=1}^\nu x_i[k] = \sum_{i=1}^\nu x_i[0] \\ x_j[k] \in [L_\mu, L_{\mu+1}] \end{cases} \quad \forall k \geq k_0, \; \forall j \in \boldsymbol{\nu} \quad (28)$$

or:

$$\begin{cases} \sum_{i=1}^\nu x_i[k] = \sum_{i=1}^\nu x_i[0] \\ x_j[k] \in (\bar{L}_\mu, \bar{L}_{\mu+1}] \end{cases} \quad \forall k \geq k_0, \; \forall j \in \boldsymbol{\nu} \quad (29)$$

Note that the above definition presents a more comprehensive description of consensus, compared to Definition 1 (see the discussion given in Remark 1). Roughly speaking,

the revised version of Definition 1 states that the consensus is reached if the numbers on the vertices of the graph ultimately lie between two consecutive levels or two consecutive splitting levels.

Revised Definition 3: Define η_1 and η_2 to be:

$$\eta_1 = \max_{i \in \mathbf{Z}} \bar{L}_i \quad \text{s.t.} \quad \bar{L}_i \leq x_{\text{ave}},$$
$$\eta_2 = \min_{j \in \mathbf{Z}} \bar{L}_j \quad \text{s.t.} \quad \bar{L}_j \geq x_{\text{ave}} \tag{30}$$

Revised Definition 4: Let $\mathcal{S}_o(\bar{L}_i)$, $i \in \mathbf{Z}$, be defined as the set of all ν-tuple $(\alpha_1, \alpha_2, ..., \alpha_\nu) \in \mathcal{S}$ such that:

$$\alpha_j \in \left(\bar{L}_i - \varepsilon(L_{i+1} - L_i), \bar{L}_i + \varepsilon(L_{i+1} - L_i) \right], \quad \forall j \in \boldsymbol{\nu} \tag{31}$$

It is noteworthy that other definitions presented in Section III carry over to the general case, such as the definitions of $\mathcal{S}$, $\mathcal{S}_o$ and $d_\varepsilon(\cdot, \mathcal{S}_o(\mu))$. Moreover, the assumption $\varepsilon \in (0, 0.5]$ remains unchanged.

One can adopt an approach similar to the one proposed earlier to prove all lemmas and theorems (presented in Section III) in the general case. This leads to the conclusions that the consensus is reached almost surely for the graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ under the SG algorithm, and that $\mathbf{x}[k]$ belongs to one of the invariant sets $\mathcal{S}_o$, $\mathcal{S}_o(\eta_1)$ or $\mathcal{S}_o(\eta_2)$, for large enough k's .

V. SIMULATION RESULTS

Consider a complete graph $\mathcal{G}$ with $\nu = 40$ and, for simplicity, assume that all edges possess the same weight equal to $\frac{2}{\nu(\nu-1)}$. Let the initial values sitting on the vertices of $\mathcal{G}$ be uniformly distributed in the box $[0, 100]^\nu$. We wish to observe how these values evolve under the quantized stochastic gossip algorithm. For this purpose, assume that the quantization is uniform, and that $\varepsilon = 0.2$. Two sets of initial states have been randomly generated, which are spelled out below:

- As the first trial, the initial values randomly generated are depicted in Figure 3. Note that the x-axis of this plot shows the index i changing from 1 to 40, and the y-axis shows the corresponding value of $x_i[0]$. The time k_1 introduced in Theorem 2 turns out to be equal to 770. The final values at this time are plotted in Figure 4. Since these numbers are spread in the interval $[52.5, 53.5]$, the point $\mathbf{x}[k_1]$ belongs to the set $\mathcal{S}_o$ (see Theorem 2). This implies that the steady-state of the vector $\mathbf{x}[k]$ is fixed, i.e. $\mathbf{x}[k] = \mathbf{x}[k_1]$, for all $k \geq k_1$. The storage function $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o)$ is sketched in Figure 5 to illustrate how it attenuates to zero in a (non-strictly) decreasing way. This is in accordance with Lemma 2.

- As the second trial, the initial values randomly generated are shown in Figure 6. The corresponding final values at the time $k_1 = 555$ are depicted in Figure 7. This figure demonstrates that $\mathbf{x}[k_1]$ belongs to the set $\mathcal{S}_o(\eta_1)$, rather than $\mathcal{S}_o$. This confirms the results of Theorem 2. Therefore, the steady state of the vector $\mathbf{x}[k]$ is not fixed, and this vector can oscillate. However,
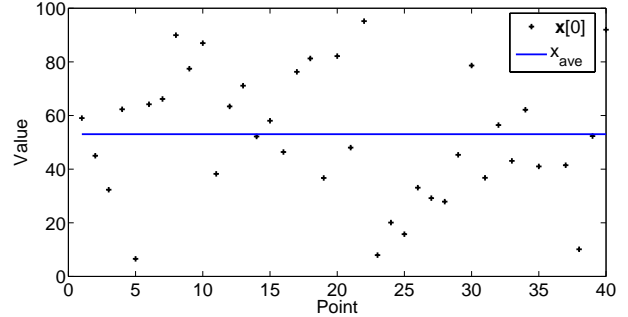

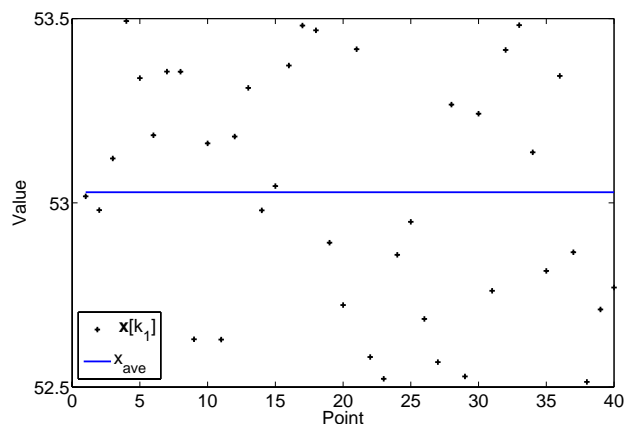Fig. 3.   The initial values on the vertices of the graph $\mathcal{G}$ for the first trial.

Fig. 4.   The final values on the vertices of the graph $\mathcal{G}$ (at time k_1) for the first trial.

$x_i[k]$, $i \in \boldsymbol{\nu}$, can take only two possible values with the difference $\varepsilon = 0.2$, in light of the definition of $\mathcal{S}_o(\mu)$. The storage function $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(\eta_1))$ is plotted in Figure 8 to illustrate the convergence rate of the SG algorithm.

VI. CONCLUSIONS

This paper deals with the distributed averaging problem over a connected weighted graph. The governing policy is that an edge of the graph is chosen at each time update with the probability equal to its weight, and then the values on its ending vertices are updated in terms of the quantized data of each other. A quantized stochastic gossip algorithm was proposed in a recent paper, which was shown to work in a particular case. In this part of the paper, it is proved that the quantized consensus is reached in the general case using this algorithm. The quantizer can be, for instance, constant or logarithmic. Some interesting steady-state properties of the numbers sitting on the vertices of the graph are derived. In the second part of the paper, the expected value of the time at which the consensus is reached will be lower and upper bounded in term of the topology of the graph, particularly its Laplacian matrix.
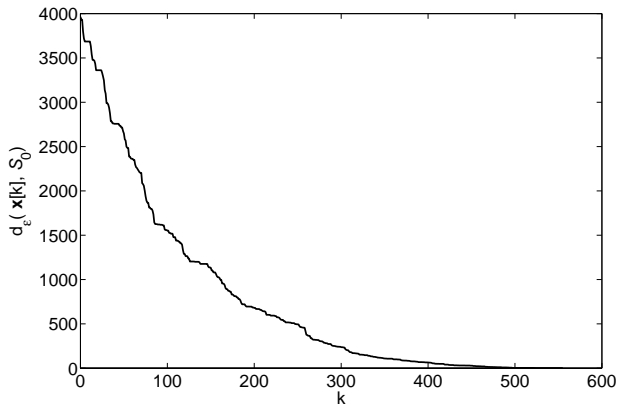
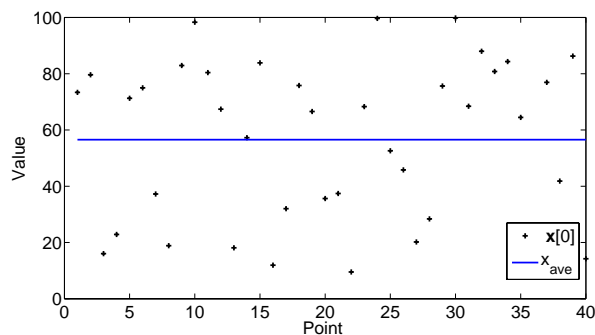Fig. 5. The storage function $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o)$ for the first trial.

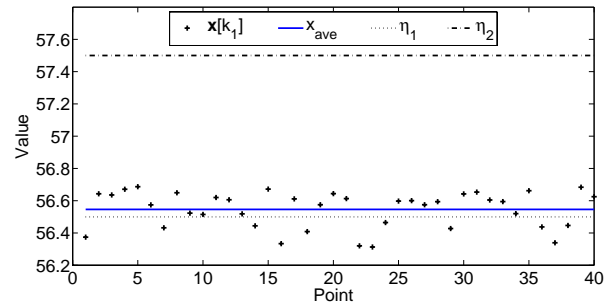Fig. 6. The initial values on the vertices of the graph $\mathcal{G}$ for the second trial.

Fig. 7. The final values on the vertices of the graph $\mathcal{G}$ (at time k_1) for the second trial.
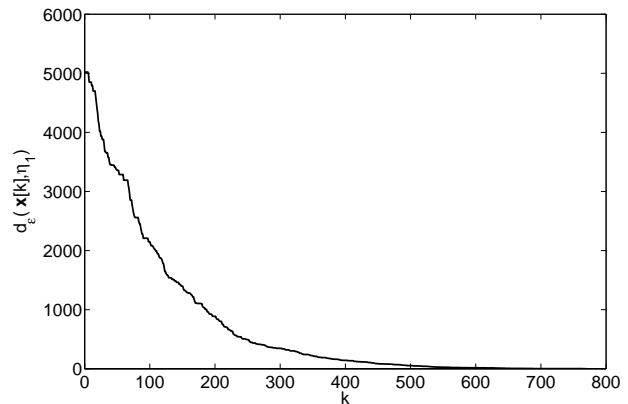

Fig. 8. The storage function $d_\varepsilon(\mathbf{x}[k], \mathcal{S}_o(\eta_1))$ for the second trial.

REFERENCES

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

[2] R. Olfati-Saber, J. A. Fax and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215-233, 2007.

[3] G. Tel, "Introduction to distributed algorithms," Cambridge University Press, 2000.

[4] N. A. Lynch, "Distributed algorithms," Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1996.

[5] D. P. Bertsekas and J. N. Tsitsiklis, "Parallel and distributed computation: Numerical methods," Belmont, MA: Athena Scientific, 1997.

[6] Y. Rabani, A. Sinclair and R. Wanka, "Local divergence of Markov chains and the analysis of iterative load-balancing schemes," in *Proceedings of IEEE Conference on Foundations of Computer Science*, 1998.

[7] Y. Kuramoto, "Chemical oscillators, waves, and turbulence," Springer-Verlag, Berlin, 1984.

[8] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, pp. 268-276, 2001.

[9] A. V. Savkin, "Coordinated collective motion of groups of autonomous mobile robots: Analysis of Vicseḱs model," *IEEE Transactions on Automatic Control*, vol. 49, no. 6, pp. 981982, 2004.

[10] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401420, 2006.

[11] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 9881001, 2003.

[12] A. Speranzon, C. Fischione and K.H. Johansson, "Distributed and collaborative estimation over wireless sensor networks," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006.

[13] J. Tsitsiklis, "Problems in decentralized decision making and computation," PhD thesis, Dept. of Electrical Engineering and Computer Science, M.I.T., Boston, MA, 1984.

[14] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah, "Analysis and optimization of randomized gossip algorithms," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, 2004.

[15] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah , "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508-2530, 2006.

[16] F. Benezit, A. G. Dimakis, P. Thiran and M. Vetterli, "Gossip along the way: Order-optimal consensus through randomized path averaging," in *Proceedings of the Allerton Conference on Communication, Control, and Computing*, 2007.

[17] A. Kashyap, T. Basara and R. Srikanta, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192-1203, 2007.

[18] P. Frasca, R. Carli, F. Fagnani and S. Zampieri, "Average consensus by gossip algorithms with quantized communication," in *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008.

[19] A. Censi and R. M. Murray, "A biologically inspired approach to real-valued average consensus over quantized channels with arbitrary deterministic accuracy," in *Proceedings of the 2009 American Control Conference*, 2009.

[20] J. Lavaei and R. M. Murray, "On quantized consensus by means of gossip algorithm – Part II: convergence time," in *Proceedings of the 2009 American Control Conference*, 2009.