Average consensus via max consensus ⋆

Riccardo Lucchese * **Damiano Varagnolo** *

* *Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, Luleå, Sweden.*
Emails: `{ riccardo.lucchese | damiano.varagnolo }@ltu.se`.

Abstract Since intuition states that it is simple and fast to compute maxima over networks, we aim at understanding the limits of computing averages over networks through computing maxima. We thus build on top of max-consensus based networks' cardinality estimation protocols a novel estimation strategy that infers averages through computing maxima of opportunely and locally generated random initial conditions. We motivate the max-consensus strategy explaining why it satisfies practical requirements, we characterize completely its statistical properties, and we analyse when and under which conditions it performs favourably against classical linear consensus strategies in static Cayley graphs.

Keywords: distributed averaging, computation of sums, order statistics

1. INTRODUCTION

Assume that each node $i = 1, \ldots, n$ of a sensor network samples a noisy measurement

$$y_i = h_i^T \theta + \nu_i \tag{1}$$

with h_i known and θ to be estimated. Distributedly computing the Least Squares (LS) estimate of θ corresponds then to evaluating

$$\widehat{\theta} = \left(\frac{1}{n} \sum_{i=1}^{n} h_i h_i^T \right)^{-1} \left(\frac{1}{n} \sum_{i=1}^{n} h_i y_i \right), \tag{2}$$

i.e., a ratio of averages of local quantities.

(2) exemplifies how certain distributed task can be solved by computing averages over networks: quoting the survey [1], many control, optimization and estimation problems such as least squares, sensor calibration, vehicle coordination and Kalman filtering can be cast as the computation of some sort of averages. In other words, average consensus represents an important tool for solving distributed tasks.

The performance of average consensus algorithms is often measured in convergence speed, i.e., the number of communication steps required to reach an agreement [2]. Indeed, the longer it takes to solve (2), the older the original information will be. There is thus a vast effort in developing "fast" average consensus strategies with provable convergence properties. Here we follow this trend, and try to understand to which extent max consensus protocols (among the fastest consensus strategies in the sense specified in Section 3) can be used for computing averages over networks.

⋆ This work is supported by the Framework Programme for Research and Innovation Horizon 2020 under grant agreement n. 636834 (DISIRE), and the Swedish research council Norrbottens Forskningsråd.

Literature review Let each node $i = 1, \ldots, n$ of the network have an initial value s_i in its memory, and assume that the aim of the nodes is to compute

$$a := \frac{1}{n} \sum_{i=1}^{n} s_i = \frac{s}{n}, \qquad s := \sum_{i=1}^{n} s_i. \tag{3}$$

The most well known and characterized average consensus approach is that of performing linear iterations of the form

$$\begin{bmatrix} a_1(k+1) \\ \vdots \\ a_n(k+1) \end{bmatrix} = P(k) \begin{bmatrix} a_1(k) \\ \vdots \\ a_n(k) \end{bmatrix}, \qquad \begin{bmatrix} a_1(0) \\ \vdots \\ a_n(0) \end{bmatrix} = \begin{bmatrix} s_1 \\ \vdots \\ s_n \end{bmatrix} \tag{4}$$

with matrices $P(k)$ consistent with the underlying graph and capturing how nodes exchange and mix their information [1]. The convergence properties of (4) depend on the spectral properties of the $P(k)$s [3], and thus on the communication topology. When the communications network can be designed, then the optimal strategy is given by a de Bruijn graph [4]. When, instead, it is given, then (for static graphs) the $P(k) = P$ leading to fastest convergence is the solution to an opportune semidefinite program [5].

Our approach to compute a in (3) is based on a different premise: instead of aggregating information through sums, we consider max-operations. Here, we first propose a max-consensus based strategy and then compare it with (4). At the best of our knowledge, there is no literature addressing these two points, while there are manuscripts describing how to compute n (and, potentially, also s) using max-operations. When quantization issues are negligible, the problem of estimating the network cardinality n through max-consensus protocols is completely solved [6]. We are, however, not aware of generalizations for estimating s and a, and not aware of solutions to estimating the cardinality n when quantization issues are considered (a first partial attempt is in [7]).

We notice that the max-consensus strategies cited above are not perfect counting mechanisms. Coupling a max-consensus-based leader election step with the classical average consensus would in fact lead to perfect counting (assuming that the leader election task terminates correctly) [8]. Nonetheless this hybrid approach has slower convergence properties (a max consensus step is followed by an average consenus step). We also notice that an alternative strategy for estimating averages is to exploit sampling-based approaches, i.e., averaging only a subset of the n original numbers $s_1, \ldots, s_n$; the quality of this approximation depends then on the empirical distribution of these quantities [9].

Assumptions Here we summarize our simplifying assumptions, omitting for brevity some basic graph-theoretic definitions (deferred to [1]).

Assumption 1. The network is represented by a static strongly connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, \ldots, n\}$ the set of nodes and $\mathcal{E}$ the set of communication links.

Assumption 2. Time is partitioned into ordered intervals indexed by $t = 0, 1, 2, \ldots$, each referred to as an "epoch". During each epoch, randomly, uniformly and i.i.d. during the epoch, each agent i in the network broadcasts its information to all its neighbors through a perfect channel (i.e., without collisions, delays, communication errors).

Assumption 3. Computations are free of quantization issues.

Assumption 4. The quantities s_i are all strictly positive.

Problem definition Given the previous assumptions, nodes can compute $m := \max_{i=1}^{n} \{s_i\}$ through iterations of the kind

$$s_i(k) = \max\left(s_i(k-1), \{s_j(k-1)\}_{j \in \mathcal{N}_i}\right), \quad s_i(0) = s_i, \tag{5}$$

with $\mathcal{N}_i$ denoting the set of neighbors of i. Protocol (5) converges to m in at most d epochs, with d the diameter of the network (notice that d can be estimated using the very same protocol [10]). In fact, the maximum m is different from the average a; nonetheless, as explained in Section 2, it is possible to modify the initial condition in (5) so that the resulting m conveys statistical information about a, eventually allowing one to compute a Maximum Likelihood (ML) estimate $\widehat{a}$ of a from m. Moreover one can improve the statistical accuracy of $\widehat{a}$ by sending more information per communication step (see Section 2).

Consider instead the classical linear average consensus protocol (4) where $P(k) = P$, consistent with the network graph and doubly stochastic, i.e., with non-negative entries and s.t. if $\mathbb{1}$ is a column vector of n ones then $P\mathbb{1} = \mathbb{1}$, $P^T\mathbb{1} = \mathbb{1}$. With these assumptions protocol (4) exponentially converges to a with rate equal to the essential spectral radius of P [1, Theorem 1].

Thus:

- the average consensus converges exponentially to a;
- the max consensus converges in d steps to $\widehat{a} \neq a$, and the estimation error can be diminished by increasing the number of scalars sent per communication step.

Choosing the Mean Squared Error (MSE) as our performance index (i.e., the sum of the squares of the local deviations from a at the generic epoch k), under certain conditions (on P, on $s_1, \ldots, s_n$, on the number of scalars used in the max consensus and others; see Section 2), the max-consensus based strategy *may* lead to better MSEs. Here, we are interested in studying when this happens.

Statement of contributions Our contributions are:

(1) derive (32), i.e., a max-consensus based ML estimator of a, and fully characterize its statistical properties in (37) and (38);
(2) motivate estimator (32) as the unique possible strategy under the framework described in Section 3;
(3) characterize when, and under which conditions, estimator (32) performs better (in MSEs terms) than the average-consensus strategy (4) when considering Cayley graphs.

Structure of the manuscript Section 2 presents the estimation strategy and characterizes it from statistical perspectives. Section 3 motivates the structure of the proposed protocol from practical considerations. Section 4 compares the performance of the novel estimator with the average-consensus strategy. Section 5 concludes the manuscript with some remarks and a roadmap for future research.

2. MAX AVERAGING

We introduce and characterize an unbiased estimator of the average $a = s/n = sn^{-1}$ in (3) by means of the following 3 subsections, defining respectively a ML estimator for n^{-1} (Section 2.1), for s (Section 2.2), and for a (Section 2.3).

2.1 Estimating n^{-1}

Estimating the size of a network n has been a research topic for long. In our set-up we are interested in performing this task through max-consensus strategies under the assumption of negligible quantization effects. I.e., we assume that the memory of the generic agent i is endowed with the M_n-dimensional vector

$$y_i = [y_{i,1} \ \ldots \ y_{i,M_n}] \in \mathbb{R}^{M_n} \tag{6}$$

where each component is a real-valued scalar initialized at the origin of time as

$$y_{i,m} \sim \mathcal{U}[0,1] \text{ i.i.d.}, \quad i = 1, \ldots, n, \quad m = 1, \ldots, M_n, \tag{7}$$

and where the max-consensus communication protocol is such that for every communication epoch (cf. Assumption 2) every node updates its $y_{i,m}$'s for $m = 1, \ldots, M_n$ as

$$y_{i,m} \leftarrow \max_{j \in \mathcal{N}_i} \left(\{y_{j,m}\}, y_{i,m}\right) \tag{8}$$

so that, after at most d epochs, every $y_{i,m}$ converges to

$$y_m := \max_{j \in \mathcal{V}} \{y_{j,m}\}, \qquad m = 1, \ldots, M_n. \tag{9}$$

Let then

$$y := [y_1, \ldots, y_{M_n}]. \tag{10}$$

Using order-statistics considerations it is immediate to check that

$$p\,(y\,;\,n) = n^{M_n} \prod_{m=1}^{M_n} \left(y_m\right)^{n-1}, \tag{11}$$

so that the ML estimator of n^{-1} given y is

$$\widehat{n^{-1}} = \widehat{n^{-1}}(y) := -\frac{1}{M_n} \sum_{m=1}^{M_n} \log y_m. \tag{12}$$

This estimator, fully characterized in [6], has a probability distribution expressible in closed-form. Indeed each variable $-\log\left(y_m\right)$ is exponentially distributed with rate n; moreover the sum of M_n i.i.d. exponential random variables with rate n is a Gamma random variable with shape M_n and scale n^{-1}. $\widehat{n^{-1}}$ is thus a scaled version of this sum of exponentials

$$p\left(\widehat{n^{-1}}\,;\,n, M_n\right) = \text{Gamma}\left(M_n, (nM_n)^{-1}\right) \tag{13}$$

(M_n is the shape, $(nM_n)^{-1}$ is the scale) such that, for $M_n > 2$,

$$\mathbb{E}\left[\widehat{n^{-1}}\right] = n^{-1}, \tag{14}$$

$$\mathbb{E}\left[\left(\frac{n^{-1} - \widehat{n^{-1}}}{n^{-1}}\right)^2\right] = \text{var}\left(\frac{\widehat{n^{-1}}}{n^{-1}}\right) = \frac{1}{M_n}. \tag{15}$$

Interestingly, $\widehat{n^{-1}}$ is Minimum Variance Unbiased (MVU), i.e., efficient and it achieves its Cramér-Rao lower bound.

Remark 5. Generating $y_{i,m}$ in (7) using distributions other than the uniform does not lead to better statistical performance. Indeed by using the probability integral transform it is possible to show that generating $y_{i,m}$ using any cumulative distribution $\mathcal{P}(\cdot)$ that is absolutely continuous (the natural choice for the case considered here, where we neglect quantization issues) leads to an estimator of the form

$$\widehat{n^{-1}} = \widehat{n^{-1}}(y) := -\frac{1}{M_n} \sum_{m=1}^{M_n} \log \mathcal{P}(y_m). \tag{16}$$

The novel estimator would have the same probability density of $\widehat{n^{-1}}$ given in (13) [6, Prop. 7], and thus be statistically equivalent to the original one.

2.2 Estimating s

Estimating $s = \sum_{i=1}^{n} s_i$ can be seen as a generalization of estimating $n = \sum_{i=1}^{n} 1$, i.e., as a weighted cardinality estimation problem. In this case assume that the memory of the generic agent i is endowed with the M_s-dimensional vector

$$z_i = \left[z_{i,1}\ \dots\ z_{i,M_s}\right] \in \mathbb{R}^{M_s} \tag{17}$$

where each component is a real-valued scalar. Exploiting the fact that Beta distributions are generalizations of uniform distributions, namely,

$$u \sim \mathcal{U}\left[0,1\right] \Rightarrow u^{1/s} \sim \text{Beta}\left(s,1\right) \Rightarrow \text{Beta}\left(1,1\right) = \mathcal{U}\left[0,1\right], \tag{18}$$

we now consider the initialization of the components $z_{i,m}$ at the origin of time as

$$z_{i,m} \sim \text{Beta}\left(s_i, 1\right) \text{ i.i.d.}, \quad i = 1,\dots,n, \quad m = 1,\dots,M_s. \tag{19}$$

We thus consider the same max-consensus communication protocol as before, i.e., for each epoch every node updates every $z_{i,m}$ for $m = 1,\dots,M_s$ as

$$z_{i,m} \leftarrow \max_{j \in \mathcal{N}_i}\Big(\left\{z_{j,m}\right\}, z_{i,m}\Big) \tag{20}$$

so that, after d epochs, every $z_{i,m}$ converges to

$$z_m := \max_{j=1}^{n}\left\{z_{j,m}\right\}, \qquad m = 1,\dots,M_s. \tag{21}$$

Importantly, [11, Lemma 1] ensures that

$$z_m \sim \text{Beta}\left(\sum_{i=1}^{n} s_i, 1\right) = \text{Beta}\left(s,1\right). \tag{22}$$

Let then

$$z := \left[z_1,\dots,z_{M_s}\right]. \tag{23}$$

Since

$$p\left(z_m\,;\,n\right) = \frac{1}{\text{B}\left(s,1\right)} z_m^{s-1} = s z_m^{s-1} \tag{24}$$

where $\text{B}\left(\cdot,\cdot\right)$ is the Beta function, it follows that

$$p\left(z\,;\,s\right) = s^{M_s} \prod_{m=1}^{M_s} \left(z_m\right)^{s-1}, \tag{25}$$

so that the ML estimator of s given z is structurally the inverse of (12), i.e.,

$$\widehat{s}_{\text{ML}} = \widehat{s}_{\text{ML}}(z) := \frac{M_s}{-\sum_{m=1}^{M_s} \log z_m}. \tag{26}$$

Since the ML estimator $\widehat{s}_{\text{ML}}$ is biased (see, e.g., [6, Sec. III]), we introduce its unbiased version

$$\widehat{s} = \widehat{s}(z) := \frac{M_s - 1}{-\sum_{m=1}^{M_s} \log z_m}. \tag{27}$$

$\widehat{s}$ shares similar properties with $\widehat{n^{-1}}$:

$$p\left(\widehat{s}\,;\,s, M_s\right) = \text{Inv-Gamma}\left(M_s, s(M_s - 1)\right) \tag{28}$$

from which it follows, for $M_s > 2$,

$$\mathbb{E}\left[\widehat{s}\right] = s, \tag{29}$$

$$\mathbb{E}\left[\left(\frac{s - \widehat{s}}{s}\right)^2\right] = \text{var}\left(\frac{\widehat{s}}{s}\right) = \frac{1}{M_s - 2}. \tag{30}$$

Remark 5 is valid also for $\widehat{s}$; i.e., generating $z_{i,m}$ using other absolutely continuous cumulative distributions rather than the uniform one does not lead to performance improvements. Moreover $\widehat{s}$ exploits the same complete and sufficient statistic exploited by $\widehat{n}$, and is thus MVU as well.

2.3 Estimating a

Having computed the ML estimators for n^{-1} and s is instrumental for computing the ML estimator for the average a. Indeed, the ML estimator for a is the composition of the ML estimators for s and n^{-1}:

Lemma 6. Assume that the nodes have already reached consensus on y and z in (10) and (23) respectively. Then

$$\arg\max_{\widetilde{a} \in \mathbb{R}} p\left(y, z\,;\,\widetilde{a}\right) = \widehat{s}_{\text{ML}}(z)\widehat{n^{-1}}(y). \tag{31}$$

The unbiased version of the ML estimator (31) is defined by

$$\widehat{a} = \widehat{a}(y,z) := \widehat{s}(z)\widehat{n^{-1}}(y). \tag{32}$$

The proof of the unbiasedness of $\widehat{a}$ exploits the independence of y and z (the latter being inherited by the fact that the $y_{i,m}$'s and the $z_{i,m}$'s are independent, and the fact that we are considering a frequentist approach where n and s are deterministic quantities). This independence implies then (for $M_n, M_s > 2$)

$$\mathbb{E}\left[\widehat{a}\right] = \mathbb{E}\left[\widehat{s}\right]\mathbb{E}\left[\widehat{n^{-1}}\right] = a \tag{33}$$

$$\begin{aligned}\mathbb{E}\left[\left(\frac{a-\widehat{a}}{a}\right)^2\right] &= \mathrm{var}\left(\frac{\widehat{a}}{a}\right) \\ &= \left(\mathrm{var}\left(\frac{\widehat{s}}{s}\right)+1\right)\left(\mathrm{var}\left(\frac{\widehat{n^{-1}}}{n^{-1}}\right)+1\right)-1 \\ &= \frac{M_n + M_s - 1}{M_n\left(M_s - 2\right)}.\end{aligned} \tag{34}$$

To reduce the notational burden, assume then $M_n + M_s =: M$ to be bounded. The natural choice for choosing M_n and M_s is then to minimize the Normalized Mean Squared Error (NMSE):

Lemma 7. Given $M > 4$, let

$$\begin{aligned}\left(M_n^*, M_s^*\right) := \arg \min_{M_n, M_s \in \mathbb{N}_+} \mathbb{E}\left[\left(\frac{a-\widehat{a}}{a}\right)^2\right] \\ \text{s.t. } M_n + M_s = M.\end{aligned} \tag{35}$$

Then

$$M_n^* = \mathrm{floor}\left(\frac{M}{2}\right) - 1 \qquad M_s^* = M - M_n^*. \tag{36}$$

For the rest of the manuscript assume that M_n and M_s have been chosen as in (36). Then, the NMSE (34) reduces to (see Figure 1)

$$\begin{aligned}\mathbb{E}\left[\left(\frac{a-\widehat{a}}{a}\right)^2\right] &= \frac{M-1}{\left(\mathrm{floor}\left(\frac{M}{2}\right)-1\right)\left(\mathrm{ceil}\left(\frac{M}{2}\right)-1\right)} \\ &= o\left(\frac{1}{M}\right).\end{aligned} \tag{37}$$

Moreover, considering that $\widehat{a}$ results from the product of an inverse gamma variate with an independent gamma variate, it follows that the distribution of $\widehat{a}$ is given by [12, Lemma 2.1]

$$\begin{aligned}p\left(\widehat{a}\ ;\ a\right) = &\frac{1}{\left(\dfrac{M_n^*}{a(M_s^*-1)}\right)^{M_s^*}\mathrm{B}\left(M_s^*, M_n^*\right)} \cdot \\ &\cdot \frac{\widehat{a}^{M_s^*-1}}{\left(1 + \dfrac{M_s^*-1}{M_n^*}a\widehat{a}\right)^{M}}\end{aligned} \tag{38}$$

As expected, Remark 5 is valid also for $\widehat{a}$; i.e., generating $y_{i,m}$ and $z_{i,m}$ using other absolutely continuous cumulative distributions rather than the uniform one does not lead to performance improvements. Moreover, since $\widehat{a}$ exploits the statistics used by $\widehat{s}$ and $\widehat{n}$, which are complete and sufficient for a, it immediately follows that $\widehat{a}$ is also MVU.

Remark 8. Max-consensus based averaging is naturally adapted to estimating generalized averages such as

$$\sqrt[\alpha]{\frac{1}{n}\sum_{i=1}^{n} s_i^{\alpha}}. \tag{39}$$

In fact, given the a priori knowledge of the exponent α, the network can exploit our protocol to distributedly generate information on the average $n^{-1}\sum_{i=1}^{n} s_i^{\alpha}$ and then infer a ML estimate of (39) as we discussed above.

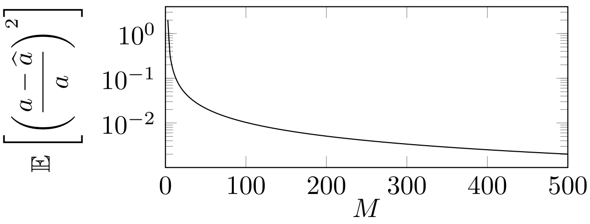

Figure 1. Graphical representation of (37), the NMSE of the estimator $\widehat{a}$ as a function of the number of scalars M transmitted during each broadcast communication.

3. MOTIVATIONS

Motivated by practical considerations, we considered the following assumptions:

A1) nodes are peers running the same information aggregation primitives, and they are not differentiated during their production process (by means, e.g., of an unique ID);

A2) time is critical, and we aim at understanding the achievable performance when the estimate is computed as soon as the information is propagated *once* from every node to every other node (i.e., in the case when the consensus is reached *as soon as possible*);

A3) there is no prior information on the s_i;

A4) the number of communicated scalars is limited, to account for finite bandwidths. (Yet we ignore quantization issues, to obtain a simplified description of trade-offs that may be encountered in real world settings; the validity of these approximations will have to be investigated in future works).

We now motivate why these assumptions lead naturally to the proposed max-consensus based algorithm.

First, Assumption A1, useful for simplifying the physical production of the nodes, suggests to use randomized algorithms. Indeed, considering deterministic initial conditions (not depending on the estimand) and deterministic aggregation mechanisms (again not depending on the estimand) would imply a non identifiability of the average. Thus randomization should act either on the initial conditions or/and in the aggregation mechanism. Since randomized aggregation strategies would violate the convergence requirement stated in Assumption A2 [1], randomization should happen when initializing the nodes' memories.

Moreover, Assumptions A2 and A3 suggest to use max-consensus protocols: indeed the convergence requirement is satisfied only by order-statistics consensus protocols, that compute the κ-th biggest (or smallest) element in the set $\{s_1, \ldots, s_n\}$. An approach may then be constructing and exchanging lists of the biggest / smallest s_i's and then infer a using L-estimators; but since we do not assume a prior

[1] We omit treating formally this issue due to space constraints, and leave it for extensions of this paper. The intuition is that if node j randomly modifies the information content of a message received by i, then i should be informed back of these changes.

on the s_i's, the performance of these estimators cannot be characterized. This leads to frequentist assumptions where random variables are constructed from the s_i's. Estimating sums using generic order-statistics on these novel r.v.s leads then to asymptotically equivalent estimators [13]; we thus choose here the simplest one, i.e., max consensus [2].

Given that we consider max-consensus protocols, Assumption A4 finally implies that we must estimate a through estimating both s and n in (3). Indeed, recalling (22), computing maxima leads to Beta random variables with a parameter given by a sum. In other words, only sums can be estimated from a Beta random variable derived from max-consensus operations. It is then clear that the generic ratio s/n cannot be estimated directly by using just *one* max consensus protocol: at least two parallel computations are needed and this motivates the structure of our estimator.

4. COMPARISON

We now compare the performance of the average-consensus protocol (4) against the ones of the max-consensus strategy (34) for Cayley graphs and different kinds conditions. We start with a general discussion of the NMSE associated to protocol 4 in Section 4.1, a general discussion on Cayley graphs in Section 4.2, and a comparison of the NMSEs of the considered protocols for Cayley graphs in Section 4.2.

4.1 Characterization of protocol (4)

Let (4) be s.t. $P(k) = P$ for every k, and let the spectrum of P be $\Lambda = \{1, \lambda_2, \ldots, \lambda_n\}$, with $1 \geq |\lambda_2| \geq \ldots \geq |\lambda_n|$. Let moreover the associated eigenvectors be $\mathbb{1}/n, v_2, \ldots, v_n$, normalized so that $\|v_i\|_2 = 1$, $i = 2, \ldots, n$. Consider then the notation $\boldsymbol{s} := [s_1, \ldots, s_n]^T$ and $\boldsymbol{a}(k) := [a_1(k), \ldots, a_n(k)]^T$, so that (4) reduces to $\boldsymbol{a}(k) = P\boldsymbol{a}(k-1)$, $\boldsymbol{a}(0) = \boldsymbol{s}$. With this notation, $a = \mathbb{1}^T\boldsymbol{s}/n$; we can thus define the NMSE associated to P and $\boldsymbol{s}$ at time k as

$$\mathrm{NMSE}\left(\boldsymbol{a}(k)\right) := \frac{\|\boldsymbol{a}(k) - \mathbb{1}a\|^2}{\|\mathbb{1}a\|^2} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{a_i(k) - a}{a}\right)^2.$$

The aim is then to compare $\mathrm{NMSE}\left(\boldsymbol{a}(k)\right)$ with $\mathbb{E}\left[\left(\frac{a-\widehat{a}}{a}\right)^2\right]$ in (37), i.e., the average of the local normalized squared errors induced by the average consensus in a generic epoch k with the expected normalized squared error of the max consensus *assuming that this has converged* (in other words, for $k \geq d$).

Instrumental to this comparison, we decompose the vector $\boldsymbol{s}$ in two components, one parallel to $\mathbb{1}$ and one orthogonal to it. I.e., we let

$$\boldsymbol{s} = \boldsymbol{s}^{\parallel} + \boldsymbol{s}^{\perp}, \quad \boldsymbol{s}^{\parallel} := \frac{\mathbb{1}\mathbb{1}^T}{n}\boldsymbol{s} = \mathbb{1}a, \quad \boldsymbol{s}^{\perp} := \boldsymbol{s} - \boldsymbol{s}^{\parallel}, \quad (40)$$

so that, since $P\mathbb{1} = \mathbb{1}$,

$$\boldsymbol{a}(k) = P^k\boldsymbol{s} = \boldsymbol{s}^{\parallel} + P^k\boldsymbol{s}^{\perp} = \mathbb{1}a + P^k\boldsymbol{s}^{\perp}. \quad (41)$$

Thus, given the spectral decomposition of P,

$$\|\boldsymbol{a}(k) - \mathbb{1}a\|^2 = \left\|P^k\boldsymbol{s}^{\perp}\right\|^2 = \left\|\sum_{i=2}^{n}\lambda_i^k\left(v_i^T\boldsymbol{s}^{\perp}\right)v_i\right\|^2. \quad (42)$$

[2] We nonetheless notice that using generic order-statistic consensus strategies is better when the size of the network is small [13].

Assume now that nodes start from a given “dissensus” level

$$\|\boldsymbol{s} - \mathbb{1}a\| = \varphi > 0, \quad (43)$$

and that for simplicity $\boldsymbol{s}^{\perp} = \varphi v_i$ for an opportune $i = 2, \ldots, n$. Thus

$$\mathrm{NMSE}\left(\boldsymbol{a}(k)\right) = \frac{\varphi^2}{na^2}\lambda_i^{2k}, \quad (44)$$

i.e., the best convergence is achieved for $\boldsymbol{s}^{\perp} \parallel v_n$, while the worst is for $\boldsymbol{s}^{\perp} \parallel v_2$ (the very well known fact that the convergence rate of (4) is asymptotically dominated by λ_2, the essential spectral radius of P).

4.2 Essentials on Cayley graphs

We notice that the problem of selecting the P leading to the fastest convergence properties can be framed in terms of an opportune semi-definite program [5]. Here, we focus on Cayley graphs because of the availability of bounds on the essential spectral radius of the P associated to a generic graph in this class [3].

We recall that a Cayley graph $\mathcal{G}(X, S)$, where X is a finite Abelian group of order $\|X\| = n$ and $S \subseteq X$, is a graph with vertex set $V = G$ and edge set $E = \{(x_1, x_2) \in X \times X : x_1 - x_2 \in S\}$. If S generates X then $\mathcal{G}(X, S)$ is strongly connected. If S contains all the inverses of its elements then the associated Cayley graph is undirected. A matrix P is then called a Cayley matrix if there exists a function $\pi : G \mapsto \mathbb{R}$ such that $[P_{ij}] = \pi(i-j)$ (with i and j denoting both the i-th and j-th element of X respectively and the i-th row and j-th column of P). A stochastic Cayley matrix P is also doubly stochastic, i.e., $P\mathbb{1} = \mathbb{1}$ implies $\mathbb{1}^TP = \mathbb{1}^T$. An important result is the following (tight) bound [3]:

Theorem 9. Let X be a finite Abelian group of order n and S be a subgroup of G containing zero. Then there exists a positive constant $c \leq 2\pi^2$, independent of X and S, such that for all stochastic P consistent with $\mathcal{G}(X, S)$ there holds

$$\rho(P) \geq 1 - \frac{c}{n^{2/(\|S\|-1)}}, \quad (45)$$

with $\rho(\cdot) : \mathbb{R}^{n\times n} \mapsto \mathbb{R}$ being the essential spectral radius.

This means that even if P has an optimal $\rho(P)$, then its slowest mode of convergence cannot be faster than a certain quantity depending on the size and the number of communication links of the network.

Then, as long the analysis is restricted to the slowest mode of convergence, since (37) is bounded above by $4/(M-2)$, Theorem 9 and (44) give the sufficient condition

$$M \geq \frac{4na^2}{\varphi^2}\left(1 - \frac{2\pi}{n^{2/(\|S\|-1)}}\right)^{-2d} + 2 \quad (46)$$

ensuring for which M the NMSE of the max-consensus strategy is better than the one of the classical average consensus protocol.

4.3 An example

Consider the group $X = \mathbb{Z}_n$, the generators $S = \{0, 1\}$, and the associated Cayley graph $\mathcal{G}(X, S)$. For this network it can be shown that the optimal P in given by

$$P_{i,j} = \begin{cases} \frac{1}{2} & \forall (i,j) \in E \\ 0 & \text{otherwise.} \end{cases} \tag{47}$$

and thus the essential spectral radius is

$$\rho(P) = \left(\frac{1}{2} + \frac{1}{2} \cos\left(\frac{2\pi}{n} \right) \right)^{1/2}. \tag{48}$$

The NMSE performance of averaging through our max- and average- consensus protocols for this network are compared in Figure 2.

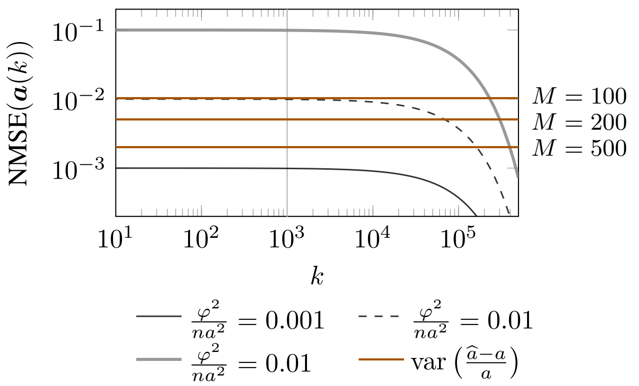

Figure 2. Graphical comparison of the NMSE in (44) against (34) for the network with Cayley graph of Section 4.3, $n = 1000$ and for different values of the initial dissensus $\frac{\varphi^2}{na^2}$ and the number of scalars M.

5. CONCLUDING REMARKS

Averaging over networks is a basic tool for distributed computations. In practice, it is important that averaging protocols have fast dynamics and it is thus interesting to study how averaging can be implemented on top of fast aggregation schemes such as max consensus.

The possibility of estimating networks cardinalities with max consensus protocols is suggestive of the possibility of estimating averages using max operations. To the best of our knowledge, here we formally propose a novel mechanism (stemming from the specific assumptions considered in Section 3) for estimating averages on top of the aggregation maxima. We characterized the statistical performance of the novel estimator and started considering when it performs better than linear consensus strategies.

Unfortunately, due to the lack of tight bounds describing the essential spectral radius of a generic average consensus matrix P, it proves difficult to solve the problem of selecting the best performing strategy. Instead, we derived a characterization for Cayley graphs, obtaining (46), i.e., an analytical sufficient condition ensuring when the performance of averaging via max-consensus are better then those of the classical average consensus in terms of the Normalized Mean Squared Error (NMSE) (assuming the worst case dynamics). As expected, there is no uniformly-better strategy: depending on the initial condition, either the first or the latter wins.

There are several open questions to be studied: first, how the estimator is affected by quantization effects; second, how the sufficient condition (46) translates for more general graphs; and third, how a Bayesian prior on the s_i's can be encoded in the estimation strategy.

REFERENCES

[1] F. Garin and L. Schenato, "A survey on distributed estimation and control applications using linear consensus algorithms," in *Networked Control Systems*. Springer, 2011, vol. 406, ch. 3, pp. 75–107.

[2] R. Carli, F. Garin, and S. Zampieri, "Quadratic indices for the analysis of consensus algorithms," in *Information Theory and Applications Workshop, ITA 2009*, 2009, pp. 96–104.

[3] R. Carli, F. Fagnani, A. Speranzon, and S. Zampieri, "Communication constraints in the average consensus problem," *Automatica*, vol. 44, no. 3, pp. 671–684, 2008.

[4] J. C. Delvenne, R. Carli, and S. Zampieri, "Optimal strategies in the average consensus problem," *Systems and Control Letters*, vol. 58, no. 10-11, pp. 759–765, 2009.

[5] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, no. 1, pp. 65–78, Sept. 2004.

[6] D. Varagnolo, G. Pillonetto, and L. Schenato, "Distributed cardinality estimation in anonymous networks," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 645–659, 2014.

[7] R. Lucchese, D. Varagnolo, J.-C. Delvenne, and J. Hendrickx, "Network cardinality estimation using max consensus: the case of Bernoulli trials," in *IEEE Conference on Decision and Control (submitted)*, 2015.

[8] I. Shames, T. Charalambous, C. N. Hadjicostis, and M. Johansson, "Distributed Network Size Estimation and Average Degree Estimation and Control in Networks Isomorphic to Directed Graphs," in *Allerton Conference on Communication Control and Computing*, 2012.

[9] S. Bodas and D. Shah, "Fast averaging," in *IEEE International Symposium on Information Theory - Proceedings*, 2011, pp. 2153–2157.

[10] F. Garin, D. Varagnolo, and K. H. Johansson, "Distributed estimation of diameter, radius and eccentricities in anonymous networks," in *3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems*, 2012.

[11] R. Cohen, L. Katzir, and A. Yehezkel, "A unified scheme for generalizing cardinality estimators to sum aggregation," *Information Processing Letters*, vol. 115, no. 2, pp. 336–342, 2015.

[12] M. M. Ali, M. Pal, and J. Woo, "On the Ratio of Inverted Gamma Variates," *Austrian Journal of Statistics*, vol. 36, no. 2, pp. 153–159, 2007.

[13] R. Lucchese and D. Varagnolo, "Networks cardinality estimation using order statistics," in *American Control Conference*, 2015.