Consensus Algorithms with State-dependent Weights

Ondrej Slučiak, Member IEEE, and Markus Rupp, Senior Member IEEE

***Abstract*—We provide an analysis of a distributed consensus-type algorithm with weights dependent only on the received data. Our approach does not require any global knowledge of the network which distinguishes our approach from most other contributions. In general, the weights may be further modified by any function. We provide general convergence conditions, prove several properties, derive steady states for such locally derived weights, as well as analyze the convergence rate of the algorithm.**

***Index Terms*—consensus, weights, generalized means, convergence, distributed algorithms**

I. Introduction

Typically, in distributed consensus problems [2], the mixing weights are designed based on parameters that depend on the network topology, such as node degrees or network size [3].

Contribution and previous work.: Although some work bypassing this requirement exists, e.g., [3]–[5], we here provide a novel general method for selecting weights based only on the (local) *received data*, without any (global) knowledge of the network. Note, e.g., that in [6], [7] nonlinear functions were applied on the data, nevertheless, the weights remained fixed (or they were switching between two weight models) and dependent on the topology of the network. We prove that distributed algorithms with weights dependent on the states converge to *a consensus* even for a time-variant, so-called switching topology, i.e., a topology where links may appear and disappear. We further conjecture (and provide an example) that under some conditions the weights may adopt negative values. Note that this conjecture is in agreement with findings from [8], [9] where negative weights also led to successful convergence. The algorithms in [3], [5] follow from our model as a special case. By proper selection of the weights we provide an algorithm which converges to a harmonic mean of the initial values (unlike classical *average* consensus). For several other special cases we derive bounds on the steady states. Furthermore, our contribution includes novel bounds on generalized (Lehmer) means and we discuss their relation to consensus algorithms. Convergence rate of the state-dependent consensus algorithms is also analyzed.

We conclude that the functions applied on the received data influence the steady state of the algorithm. This motivates applications for which, unlike classical *average* consensus algorithms [2]–[4] where the algorithm converges to the average of the initial values, the average is not required (or undesirable), but *an agreement* is sufficient [5], [10], [11].

Ondrej Slučiak and Markus Rupp are with the Institute of Telecommunications, Vienna University of Technology, A-1040 Vienna, Austria (e-mail: {ondrej.sluciak,markus.rupp}@nt.tuwien.ac.at). This work was supported by the Austrian Science Fund (FWF) under project grants S10611-N13 within the National Research Network SISE. Preliminary parts of this work were previously published at ICASSP, Florence, Italy, May 2014 [1].

Notation: The number of nodes (network size) is denoted by N. The set of neighbors of a node i is denoted as $\mathcal{N}_i$. If node i is included in its neighborhood (there is a self-loop), we write $\mathcal{N}_i^+ \triangleq \mathcal{N}_i \cup \{i\}$. We always assume to have a network described by an undirected graph (bidirectional links). A degree of a node i (number of neighbors) is denoted as $d_i \triangleq |\mathcal{N}_i|$, $\mathbf{I}$ is an identity matrix and $\mathbf{1}$ is a (column) vector of all ones. A row vector is denoted as $\mathbf{x}^\top$. Elementwise product of two matrices (vectors) is denoted by $\circ$. Logarithm of each element in a vector is denoted by $\log \mathbf{x}(k) = (\log x_1(k), \log x_2(k), \dots, \log x_N(k))$.

We further assume synchronous updates of the network, i.e., the iteration (time) index k is common to all nodes in the network.

In Sec. II we define the consensus algorithm with state-dependent weights. In Sec. III we analyze the general properties of the proposed algorithm, review several mean functions, prove novel upper bound on the so-called Lehmer mean, and discuss the relation of quasi-arithmetic means to our state-dependent consensus algorithm. In Sec. IV we analyze the steady states of our consensus algorithm in case of several special weight functions. In Sec. V we analyze the convergence rate. In Sec. VI we compare the theory with simulations. Sec. VII concludes the paper.

II. Consensus algorithm with state-dependent weights

In this section we define a general consensus algorithm with weights dependent on the received data. The received data may be further modified by some functions. As we prove, for a broad class of general functions, the algorithm converges to a consensus.

Theorem 1. *Assuming a static network, then for any set of initial numbers $x_i(0) \in \mathbb{R}$, ($i = 1, 2, \dots, N$), the update*

$$x_i(k) = w_{ii}(k-1)x_i(k-1) + \sum_{j \in \mathcal{N}_i} w_{ij}(k-1)x_j(k-1), \quad (1)$$

with weights

$$w_{ij}(k) = \frac{f_j(x_j(k); k)}{f_i(x_i(k); k) + \sum_{j' \in \mathcal{N}_i} f_{j'}(x_{j'}(k); k)}, \quad (2)$$

for any function $f_i(\cdot; k) : \mathbb{R} \to \mathbb{R}_0^+ (\mathbb{R}_0^-)$; with the convention that if for finite times $\{k_0\}$, $f_j(x_j(k_0); k_0) = 0, \forall j \in \mathcal{N}_i^+$, then $w_{ii}(k_0) = 1$; Eq. (1) *asymptotically converges to a consensus, i.e.,*

$$\lim_{k \to \infty} x_i(k) = c, \quad \forall i = 1, 2, \dots, N.$$

Remark: Notice that if we violate the condition that $f_j(x_j(k_0); k_0) = 0, \forall j \in \mathcal{N}_i^+$ happens only *finite* many times (k_0 is from a finite set), then $w_{ii}(k) = 1$ and $x_i(k) = x_i(k_0),\ \forall i; k > k_0$. In other words, the consensus can be reached, in this case, only if $|x_i(k_0) - c| < \epsilon$, for all i and some $\epsilon > 0$, already holds (algorithm has already converged, when we disconnect some nodes). However, if this happens only a finite number of times, the information can still spread over the network and a global consensus may be achieved.

Note that by a *static* network in Theorem 1 we imply that its adjacency matrix $\mathbf{A}$ does not vary (see Eq. (3) ahead), i.e., nodes cannot arbitrarily create new links to other nodes, the size of network is fixed, i.e., the set of neighbors of each node $\mathcal{N}_i$ is constant. However, as noted in the remark, allowing some weights to become zero at a finite amount of time is equivalent to a network with switching topology (links between the nodes may vary).

Still having in mind that the functions at the nodes $f_i(\cdot;k)$ may change in time k, for the sake of clarity, we drop the additional argument k in the following notation, i.e., $f_i(\cdot)$, where this property is not explicitly required.

In the following proof we state several important terms and equations which we use further in the next sections.

Proof of Theorem 1. Since all functions $f_i(\cdot):\mathbb{R}\rightarrow\mathbb{R}_0^+(\mathbb{R}_0^-)$ $(i = 1, 2, \dots, N)$, we conclude that $0 \le w_{ij}(k) \le 1, \forall i, j$ and from a global (network) point of view, we observe that the weight matrix (2), i.e.,

$$\mathbf{W}(k) = [\mathbf{W}(k)]_{ij} \equiv w_{ij}(k),$$

has the following properties:

1) $\mathbf{W}(k)\mathbf{1} = \mathbf{1}$ for all k, ($\mathbf{W}(k)$ is row-stochastic)
2) eigenvalue $\lambda_{\max}(k) = 1$, with corresponding right eigenvector $\mathbf{v}_{\max}(k) = \mathbf{1}$,
3) left eigenvector corresponding to the eigenvalue $\lambda_{\max}(k)$, $\mathbf{u}_{\max}^\top(k) = (u_1(k), \dots, u_i(k), \dots, u_N(k))$ with

$$u_i(k) = \frac{f_i(x_i(k))\Big(f_i(x_i(k)) + \sum\limits_{j\in\mathcal{N}_i} f_j(x_j(k))\Big)}{\sum\limits_{i'=1}^{N} f_{i'}(x_{i'}(k))\left(f_{i'}(x_{i'}(k)) + \sum_{j'\in\mathcal{N}_{i'}} f_{j'}(x_{j'}(k))\right)}.$$

The choice of the weights (2) thus guarantees the well-known necessary conditions for convergence [8], [12] of Eq. (1), i.e., $\lim_{k\to\infty}\mathbf{x}(k) = \lim_{K\to\infty}\prod_{k=0}^{K}\mathbf{W}(k)\mathbf{x}(0)$ may exist.

Furthermore, notice that matrix $\mathbf{W}(k)$ can be decomposed into

$$\mathbf{W}(k) = \mathbf{D}_1(k)(\mathbf{I} + \mathbf{A})\mathbf{D}_2(k), \tag{3}$$

where $\mathbf{D}_1(k) = \begin{pmatrix} \frac{1}{\sum_{j\in\mathcal{N}_1^+} f_j(x_j(k))} & & \\ & \frac{1}{\sum_{j\in\mathcal{N}_2^+} f_j(x_j(k))} & \\ & & \ddots \end{pmatrix}$, $\mathbf{D}_2(k) = \begin{pmatrix} f_1(x_1(k)) & & \\ & f_2(x_2(k)) & \\ & & \ddots \end{pmatrix}$, and $\mathbf{A}$ is the adjacency matrix of the network. Thus, in case matrix $\mathbf{A}$ is symmetric (undirected graph), any matrix of form (3) can be regarded as a transition matrix of a *reversible* (homogeneous) Markov chain [13], [14]. Thus, if $\mathbf{W}(k)$ is constant for all $k > 0$, we find that $\lim_{K\to\infty}(\mathbf{W}(k))^K = \overline{\mathbf{W}}(k) = \mathbf{v}_{\max}(k)\mathbf{u}_{\max}^\top(k)$.

However, in general, we are interested in the convergence of the time-varying matrices $\mathbf{W}(k)$, i.e.,

$$\lim_{K\to\infty}\mathbf{W}(K)\mathbf{W}(K-1)\dots\mathbf{W}(0) = \lim_{K\to\infty}\prod_{k=0}^{K}\mathbf{W}(k). \tag{4}$$

From the theory of Markov chains, we call a sequence of matrices *strongly ergodic*, if the product of transition matrices $\mathbf{W}(k)$ converges to a constant matrix with identical rows[1]. If a product of transition matrices $\mathbf{W}(k)$ does not converge to a fixed matrix, but as $k \to \infty$ the difference between rows decreases, such process is called *weakly ergodic* [17], [18]. In other words, ergodic means that the states $x_i(k)$ "forget" their past. From mathematical point of view, for any initialization of $x_i(0)$, after some time, all states $i \in \{1, 2, \dots, N\}$ converge to the same value (regardless from their initial state).

To prove Theorem 1, we follow standard mathematical proofs and utilize the notion of *coefficient of ergodicity* $\mu(\mathbf{W})$ as proposed in [18]. Here, we use a generalized form as introduced in [19], i.e.,

$$\mu(\mathbf{W}) \triangleq \min_{i,i'}\|\min(w_{ij}, w_{i'j})\|_1 = \min_{i,i'}\sum_j \min(w_{ij}, w_{i'j}).$$

Thus, this coefficient "measures" the similarity between the rows of a matrix $\mathbf{W}$. Clearly, $\mu(\mathbf{W}) = 1$ if and only if all the rows of a row-stochastic matrix $\mathbf{W}$ are identical[2]. From [18, Lemma 1], we further know that for a product of two *row-stochastic matrices* $\mathbf{A}$ and $\mathbf{B}$, i.e., $\mathbf{C} = \mathbf{AB}$, then $\mu(\mathbf{C}) \ge \mu(\mathbf{A})$, thus the coefficient of ergodicity $\mu(\cdot)$ is non-decreasing with the product of matrices. Thus, to prove the Theorem 1, it would be sufficient to show $\lim_{K\to\infty}\mu(\prod_{k=0}^{K}\mathbf{W}(k)) = 1$. However, to simplify the proof, we introduce yet another coefficient of ergodicity [18], [19], i.e.,

$$\delta(\mathbf{W}) \triangleq \max_{i,i'}\|\mathbf{w}_i^\top - \mathbf{w}_{i'}^\top\|_\infty = \max_j\max_{i,i'}|w_{ij} - w_{i'j}|,$$

where $\mathbf{w}_i^\top$ is the i-th row of matrix $\mathbf{W}$. Similarly to $\mu(\mathbf{W})$, $\delta(\mathbf{W})$ describes the difference between the rows of a matrix $\mathbf{W}$. It can be observed that $\delta(\mathbf{W}) = 0$ if and only if all the rows of $\mathbf{W}$ are identical.

From [18, Lemma 3], we know that for a product of two row-stochastic matrices $\mathbf{A}$ and $\mathbf{B}$, i.e., $\mathbf{C} = \mathbf{AB}$, it holds that

$$\delta(\mathbf{C}) \le (1 - \mu(\mathbf{A}))\delta(\mathbf{B}).$$

By extending this to an infinite product of row-stochastic matrices, we obtain [18, Theorem 2]

$$\delta\left(\prod_{k=0}^{\infty}\mathbf{W}(k)\right) \le \prod_{k=0}^{\infty}(1 - \mu(\mathbf{W}(k))).$$

For the proof we further require the notion of so-called *scrambling* matrices as defined in [18]. A scrambling matrix is a matrix where for every pair of rows, say (i, i'), there exists at least one column, say j, such that both $w_{ij} > 0$ and $w_{i'j} > 0$. Alternatively, we can say that no two rows of matrix $\mathbf{W}$ are

[1]Note that unlike non-homogeneous Markov chains where the *right* (forward) product of transition matrices is considered, we are interested in the *left* (backward) product of the matrices. The theorems and tools used in the proof here, however, do not depend on the order of the matrices [15], [16].

[2]In general, if all rows of a matrix $\mathbf{W}$ sum up to r, $\mu(\mathbf{W}) = r$.

orthogonal [20]. Note that a scrambling matrix does not imply an irreducible matrix (connected network; see Case 3 ahead), nor the irreducible matrix imply scrambling. Also note that any matrix with a positive column is a scrambling matrix.

It can be observed that for any scrambling matrix $\mathbf{W}$, $0 < \mu(\mathbf{W}) < 1$. And, $\mu(\mathbf{W}) = 0$, if and only if $\mathbf{W}$ is not a scrambling matrix. Also, as mentioned before $\mu(\mathbf{W}) = 1$ if and only if all the rows of a row-stochastic matrix $\mathbf{W}$ are identical. It thus suffices to show that there are infinitely many scrambling matrices in the product $\prod_{k=0}^{\infty} \mathbf{W}(k)$.

To prove Theorem 1, for the general case $f_i(\cdot) \geq 0$ (non-negative), we have to distinguish the following three cases:

Case 1) If all $f_i(x_i(k)) > 0$ (strictly positive) (or all $f_i(\cdot)$ are strictly negative), we observe that matrix $\mathbf{W}(k)$ is a *primitive* matrix (irreducible since it remains connected; and aperiodic since at least one self-loop will be present) [21]. We further observe from decomposition (3) that *any product* of $\mathbf{W}(k)\mathbf{W}(k-1)$ is again a primitive matrix[3]. Thus, for any positive diagonal matrix $\tilde{\mathbf{D}}(k) = \mathrm{diag}(\frac{f_1(x_1(k))}{\sum_{j\in\mathcal{N}_1^+} f_j(x_j(k-1))}, \dots)$, matrix $(\mathbf{I}+\mathbf{A})\tilde{\mathbf{D}}(k)(\mathbf{I}+\mathbf{A})$ eventually becomes a scrambling matrix (information from some node spreads to all other nodes after some iterations, i.e., there will be a positive column). Such "sub-products" of scrambling matrices appear in (4) infinitely many times.

Case 2) If we at some times $\{k_0\}$ disconnect all neighboring nodes of node i, node i stops receiving data (becomes isolated), i.e., $f_j(x_j(k_0)) = 0, \forall j \in \mathcal{N}_i^+$, then $w_{ii}(k_0) = 1$, and the value at node i remains constant, i.e., $x_i(k_0+1) = x_i(k_0)$. In this case $\mu(\mathbf{W}(k_0)) = 0$. Thus, clearly, if we did not have the condition that this case happens only finite many times, $\lim_{K\to\infty} \delta\left(\prod_{k=0}^{K} \mathbf{W}(k)\right)$ could be larger than 0, and thus a consensus might not be reached.

Case 3) We disconnect at some times $\{k_1\}$ some (at least one, but not all) of neighboring nodes of node i, i.e., $f_j(x_j(k_1)) = 0$, for some $j \in \mathcal{N}_i^+$. This means that the j-th columns of $\mathbf{W}(k_1)$ will be equal to $\mathbf{0}$. Matrix $\mathbf{W}(k_1)$ nevertheless remains row-stochastic with a positive off-diagonal element in every row. Note that in this case the diagonal element $w_{ii}(k_1) = 0$, thus this generalizes the conditions on convergence as proposed in [5], [24]. Also note, that this case may happen infinitely many times ($\forall k$).

Case 3 represents the case when some nodes at times $\{k_1\}$ do not transmit anything, only receive (recall, that they are not isolated), and thus the network is not (strongly) connected anymore. Similarly to Case 1, after some (finite) steps $|\mathcal{I}_i|$ (depending on the connectivity of the network) there will be a strictly positive column in the matrix $\overline{\mathbf{W}}^{(\mathcal{I}_i)} \triangleq \prod_{k\in\mathcal{I}_i} \mathbf{W}(k)$, and thus $\overline{\mathbf{W}}^{(\mathcal{I}_i)}$ will be a scrambling matrix. Set $\mathcal{I}_i$ contains iterations indices of the i-th "sub-product". For (infinitely) many iterations, $\{k_1\} = \mathcal{I}_1 \cup \mathcal{I}_2 \cup \dots$, $\mu(\overline{\mathbf{W}}^{(\mathcal{I}_i)}) \leq 1$ $(\forall i)$, and thus

$$\delta\left(\prod_{k=0}^{\infty} \mathbf{W}(k)\right) \leq \prod_{k=0}^{\infty}(1-\mu(\mathbf{W}(k))) = \prod_{k\in\{k_0\}}(1-\mu(\mathbf{W}(k))) \prod_{k\in\{k_1\}}(1-\mu(\mathbf{W}(k))) = \prod_{k\in\{k_0\}} \underbrace{(1-\mu(\mathbf{W}(k)))}_{=1} \prod_{i=1}^{\infty} \underbrace{(1-\mu(\overline{\mathbf{W}}^{(\mathcal{I}_i)}))}_{<1} = 0.$$

This concludes the proof of Theorem 1. ∎

For the sake of completeness, let us briefly compare the non-homogeneous Markov chains and the consensus updates in the case of time-varying functions $f_i(x_i(k);k)$ and the case when functions $f_i(x_i(k))$ depend only on the input value.

If we allow functions $f_i(x_i(k);k)$ to be different for any k, from the theory Markov chains we know that the right (forward) product of the weight matrices $\mathbf{W}(k)$, i.e., $\mathbf{W}(0)\mathbf{W}(1)\dots\mathbf{W}(K)$, may not tend to a fixed consensus matrix. Nevertheless, the nodes reach the same value (consensus), since the sequence of the matrices is *weakly ergodic*. Clearly, this holds even if the functions $f_i(x_i(k);k)$ are random or randomly varying, independently from the input value $x_i(k)$. In case we have fixed functions $f_i(x_i(k))$, we observe that the right product of matrices converge to a fixed consensus matrix. Thus, such process is *strongly ergodic*. Moreover, if the functions are node independent, i.e., $f(x_i(k))$, then[4] $\lim_{k\to\infty} \mathbf{W}(k) = (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})$.

In our case, the situation simplifies, due to the fact that the left (backward) product (4) always converges to a fixed consensus matrix [15], [16], equivalently to a strongly ergodic behaviour. Thus, even if functions $f_i(x_i(k);k)$ vary with k, eventually for large K the weight matrix $\mathbf{W}(K)$ only averages a matrix with constant rows, thus leaves the product unchanged.

III. General properties

In this section we provide several properties of the consensus algorithm as defined in Theorem 1.

Lemma 1. *Consider update Eq.* (1) *with weights* (2)*, then for all k and any functions $f_i(x_i(k)) \in \mathbb{R}$, the states $x_i(k) \geq 0$, $i = 1, 2, \dots, N$,*(1) *are bounded*

$$x_{\min}(0) \leq x_i(k) \leq x_{\max}(0),$$

where $x_{\min}(0) \triangleq \min_{i=1,\dots,N} x_i(0)$ *and* $x_{\max}(0) \triangleq \max_{i=1,\dots,N} x_i(0)$.

Proof. Since (cf. Eq. (1))

$$\begin{aligned} x_i(k) &= \frac{\sum_{j\in\mathcal{N}_i^+} f_j(x_j(k-1);k-1) x_j(k-1)}{\sum_{j\in\mathcal{N}_i^+} f_j(x_j(k-1);k-1)} \\ &\leq \frac{\sum_{j\in\mathcal{N}_i^+} f_j(x_j(k-1);k-1) x_{\max}^i(k-1)}{\sum_{j\in\mathcal{N}_i^+} f_j(x_j(k-1);k-1)} = x_{\max}^i(k-1), \end{aligned}$$

[3]From [22, Lemma 4] we know that if all (long-enough) "sub-products" of product (4) are primitive matrices, then the product converges to a matrix with identical rows. Note that the condition, that also the product of matrices is a primitive matrix, is a crucial condition. In general, even if every matrix $\mathbf{W}(k)$ is primitive, the *product* of matrices need not to be a primitive matrix and thus the convergence is not satisfied (see [18] for examples).

It is worth noting that in [23] sufficient conditions on individual matrices of the product were defined: If every matrix $\mathbf{W}(k)$, $\forall k$ under independent permutations of rows or columns remains primitive, then the product converges. This condition is in our case satisfied only if $f_i(x_i(k)) > 0$, for all i, k, but it need not hold if we allow $f_i(x_i(k)) = 0$ for some i, k.

[4]It can be further shown, e.g., [25], [26, Theorem 3.4], that in this case the *right* product of the matrices converges to matrix $W^\star = \mathbf{1}\frac{1}{\sum_{i=1}^{N}(1+d_i)}(1+d_1, 1+d_2, \dots, 1+d_N)^\top$.

with $x^i_{\max}(k) \triangleq \max_{j \in \mathcal{N}_i^+} x_j(k)$.

Since $x^i_{\max}(k) \leq x_{\max}(k) \triangleq \max_{i=1,2,\ldots,N} x_i(k)$, then $x_i(k) \leq x_{\max}(k-1), (\forall i)$, and this recursively leads to the upper bound $x_i(k) \leq x_{\max}(0)$.

Similarly,

$$\begin{aligned} x_i(k) &= \frac{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1);k-1) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1);k-1)} \\ &\geq \frac{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1);k-1) x^i_{\min}(k-1)}{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1);k-1)} = x^i_{\min}(k-1), \end{aligned}$$

with $x^i_{\min}(k) \triangleq \min_{j \in \mathcal{N}_i^+} x_j(k)$.

Since $x^i_{\min}(k) \geq x_{\min}(k) \triangleq \min_{i=1,2,\ldots,N} x_i(k)$, then $x_i(k) \geq x_{\min}(k-1), (\forall i)$, and this recursively leads to the lower bound $x_i(k) \geq x_{\min}(0)$. ■

In other words, Lemma 1 states that for *any* function $f(\cdot)$, the steady state, i.e., $\lim_{k\to\infty} x_i(k)$, must lie between the minimum and maximum of the initial (non-negative) values. Thus, there exists no function (or network topology) for which the steady state of Eq. (1) would lead to, e.g., a sum of the initial values (except for the trivial case when the initial values are zero).

Lemma 2. *If the weight function $f_i(x_i(k);k) > 0$ is a convex function and $f_i(\cdot)$ is time-independent and common for all nodes, i.e., $f_i(x_i(k);k) \equiv f(x_i(k))$, then*

$$f(x_i(k)) \leq \sum_{j \in \mathcal{N}_i^+} f(x_j(k-1)),$$

and if the function is concave, then

$$f(x_i(k)) \geq \frac{\sum_{j \in \mathcal{N}_i^+} f^2(x_j(k-1))}{\sum_{j \in \mathcal{N}_i^+} f(x_j(k-1))}.$$

Proof. Lemma 2 is a straightforward result of Jensen's inequality [27], i.e., for a convex (positive) function

$$\begin{aligned} f(x_i(k)) &\leq \frac{\sum_{j \in \mathcal{N}_i^+} f(x_j(k-1)) f(x_j(k-1))}{\sum_{j \in \mathcal{N}_i^+} f(x_j(k-1))} \\ &\leq \frac{\left(\sum_{j \in \mathcal{N}_i^+} f(x_j(k-1))\right)^2}{\sum_{j \in \mathcal{N}_i^+} f(x_j(k-1))} = \sum_{j \in \mathcal{N}_i^+} f(x_j(k-1)), \end{aligned}$$

and analogously for a concave function. ■

Lemma 3. *If $f_i(x_i(k);k) = g_i(x_i(k);k) + h_i(x_i(k);k)$ (cf. Eq. (2)) such that $g_i(x_i(k);k) \geq h_i(x_i(k);k) \geq 0$, then*

$$\begin{aligned} x_i(k) &= \frac{\sum_{j \in \mathcal{N}_i^+} \big(g_j(x_j(k-1);k-1) + h_j(x_j(k-1);k-1)\big) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} \big(g_j(x_j(k-1);k-1) + h_j(x_j(k-1);k-1)\big)} \\ &\leq \frac{\sum_{j \in \mathcal{N}_i^+} g_j\big(x_j(k-1);k-1\big) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} h_j\big(x_j(k-1);k-1\big)}, \end{aligned}$$

and

$$x_i(k) \geq \frac{\sum_{j \in \mathcal{N}_i^+} h_j\big(x_j(k-1);k-1\big) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} g_j\big(x_j(k-1);k-1\big)}. \quad (5)$$

Proof. Using simple inequalities, it follows that if $f_i(x_i(k);k) = g_i(x_i(k);k) + h_i(x_i(k);k)$ such that $g_i(x_i(k);k) \geq h_i(x_i(k);k) \geq 0$, then the upper bound on the state $x_i(k)$ yields

$$\begin{aligned} x_i(k) &= \frac{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1);k-1) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1);k-1)} \\ &= \frac{\sum_{j \in \mathcal{N}_i^+} \big(g_j(x_j(k-1);k-1) + h_j(x_j(k-1);k-1)\big) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} \big(g_j(x_j(k-1);k-1) + h_j(x_j(k-1);k-1)\big)} \\ &\leq \frac{\sum_{j \in \mathcal{N}_i^+} (g_j\big(x_j(k-1);k-1\big) + g_j\big(x_j(k-1);k-1\big)) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} \big(g_j(x_j(k-1);k-1) + h_j(x_j(k-1);k-1)\big)} \\ &\leq \frac{2 \sum_{j \in \mathcal{N}_i^+} g_j\big(x_j(k-1);k-1\big) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} \big(h_j(x_j(k-1);k-1) + h_j(x_j(k-1);k-1)\big)} \\ &= \frac{\sum_{j \in \mathcal{N}_i^+} g_j\big(x_j(k-1);k-1\big) x_j(k-1)}{\sum_{j \in \mathcal{N}_i^+} h_j\big(x_j(k-1);k-1\big)}. \end{aligned}$$

The lower bound follows analogously. ■

Corollary 1. *If the function $f(\cdot)$ (Eq. (2)) is a monomial, i.e., $f_i(x_i(k);k) = x_i^p(k), p \in \mathbb{R}$, and $x_i(k) \geq 0$, then the state $x_i(k)$ is so-called Lehmer mean (see Definition 3 ahead) of the received data, i.e.,*

$$x_i(k) = \frac{\sum_{j \in \mathcal{N}_i^+} x_j^{p+1}(k-1)}{\sum_{j \in \mathcal{N}_i^+} x_j^p(k-1)}.$$

A powerful theorem which allows to bound more complex functions (see Sec. IV-D ahead) is the following.

Lemma 4. *If $h(x) = \frac{f(x)}{g(x)} : \mathbb{R}^+ \to \mathbb{R}$ is a decreasing function then for an arbitrary n*

$$\frac{\sum_{i=1}^n f(x_i) x_i}{\sum_{i=1}^n f(x_i)} \leq \frac{\sum_{i=1}^n g(x_i) x_i}{\sum_{i=1}^n g(x_i)}.$$

The inequality is reversed if $\frac{f(\cdot)}{g(\cdot)}$ is an increasing function.

Proof. See [28], [29]. ■

Moreover, for general weight functions which differ at each node i, i.e., $f_i(x_i)$, it can be shown [27, p. 313] [30] that the inequality for an arbitrary n

$$\frac{\sum_{i=1}^n f_i(x_i) x_i}{\sum_{i=1}^n f_i(x_i)} \leq \frac{\sum_{i=1}^n g_i(x_i) x_i}{\sum_{i=1}^n g_i(x_i)}$$

holds if for all x_i, x_j $(i \neq j)$ $(1 \leq i \leq n)$

$$\frac{f_i(x_i)}{f_n(x_j)} \leq \frac{g_i(x_i)}{g_n(x_j)}.$$

Based on Lemma 4 we can state the following theorem on the steady states of the consensus algorithm with state-dependent weights.

Theorem 2. *Having two consensus algorithms with state-dependent weights, i.e.,*

$$x_i(k) = \frac{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1)) x_j(k-1)}{\sum_{j \in \mathcal{N}^+} f_j(x_j(k-1))} \quad \text{(6a)}$$

and

$$\tilde{x}_i(k) = \frac{\sum_{j\in\mathcal{N}_i^+} g_j(\tilde{x}_j(k-1))\tilde{x}_j(k-1)}{\sum_{j\in\mathcal{N}^+} g_j(\tilde{x}_j(k-1))} \tag{6b}$$

where $x_i(0) = \tilde{x}_i(0)$, and assuming that functions $f_1(x), f_2(x), \ldots, g_1(x), g_2(x), \ldots$ are node independent and positive, i.e., $f(x): \mathbb{R}^+ \to \mathbb{R}$ and $g(x): \mathbb{R}^+ \to \mathbb{R}$ such that $h(x) \triangleq \frac{f(x)}{g(x)}$ is a decreasing differentiable function, and furthermore if $f(x)$ is non-decreasing such that $\frac{d}{dx}f(x) \le f(x)$, then for all i and k

$$x_i(k) \le \tilde{x}_i(k).$$

Thus, the steady state of Eq. (6a) *bounds the steady state of Eq.* (6b) *and vice versa.*

If function $h(x)$ is increasing, and $g(x)$ is non-decreasing such that $\frac{d}{dx}g(x) \le g(x)$, then $x_i(k) \ge \tilde{x}_i(k)$.

Proof. From Lemma 4 and assumption on the initial values $x_i(0) = \tilde{x}_i(0)$, it follows that $x_i(1) \le \tilde{x}_j(1)$. For $k = 2$ it follows that

$$x_i(2) - \tilde{x}_i(2) = \frac{\sum_{j\in\mathcal{N}_i^+} f(x_j(1))x_j(1)}{\sum_{j\in\mathcal{N}^+} f(x_j(1))} - \frac{\sum_{j\in\mathcal{N}_i^+} g(\tilde{x}_j(1))\tilde{x}_j(1)}{\sum_{j\in\mathcal{N}^+} g(\tilde{x}_j(1))}.$$

From [31], [32] we know that function $M(\mathbf{x}) = \frac{\sum_j f(x_j)x_j}{\sum_j f(x_j)}$ is an increasing function with respect to all x_j, i.e., if some $\tilde{x}_j \ge x_j$, then $M(\tilde{\mathbf{x}}) \ge M(\mathbf{x})$, if $f(x)$ is a non-decreasing function such that $\frac{d}{dx}f(x) \le f(x)$.

Thus, if $x_i(1) \le \tilde{x}_i(1)$, and applying Lemma 4 again, we obtain

$$\begin{aligned} x_i(2) - \tilde{x}_i(2) &= \frac{\sum_{j\in\mathcal{N}_i^+} f(x_j(1))x_j(1)}{\sum_{j\in\mathcal{N}^+} f(x_j(1))} - \frac{\sum_{j\in\mathcal{N}_i^+} g(\tilde{x}_j(1))\tilde{x}_j(1)}{\sum_{j\in\mathcal{N}^+} g(\tilde{x}_j(1))} \\ &\le \frac{\sum_{j\in\mathcal{N}_i^+} f(\tilde{x}_j(1))\tilde{x}_j(1)}{\sum_{j\in\mathcal{N}^+} f(\tilde{x}_j(1))} - \frac{\sum_{j\in\mathcal{N}_i^+} g(\tilde{x}_j(1))\tilde{x}_j(1)}{\sum_{j\in\mathcal{N}^+} g(\tilde{x}_j(1))} \\ &\le 0. \end{aligned}$$

Generalization for $k > 2$ follows straightforwardly.

The reversed case for $\frac{f(x)}{g(x)}$ being an increasing function can be proven analogously. ∎

As a consequence of Theorem 2, the steady state of one consensus algorithm can be bounded (from above and below) by another appropriate consensus algorithm. We will use this fact later on for bounding more complicated functions. Also note that the conditions on the properties of the functions are only implications, thus there may be different conditions on the weight functions which would still guarantee that the states $x_i(k)$ of one consensus algorithm are bounded by states $\tilde{x}_i(k)$ of another consensus algorithm.

A. Means

For the sake of completeness let us first recall several mean functions [27], [33] in general which we will use later on. Note that n denotes an arbitrary number and need not to be related to N.

Definition 1 (Gini mean [27], [33])**.** *The Gini mean of (positive) real numbers x_i, $i = 1, 2, \ldots, n$, is defined as*

$$K(p, q; \mathbf{x}) \triangleq \begin{cases} \left(\frac{\sum_{i=1}^n x_i^p}{\sum_{i=1}^n x_i^q}\right)^{\frac{1}{p-q}}, & \text{if } p \ne q, \\ \left(\prod_{i=1}^n x_i^{x_i{}^p}\right)^{\frac{1}{\sum_i x_i^p}} = \\ \quad = \exp\left(\frac{\partial}{\partial r}\log\sum_i x_i^r\big|_{r=p}\right), & \text{if } p = q. \end{cases}$$

where $p, q \in \mathbb{R}$.

The following two special cases of the Gini mean are of special interest for us.

Definition 2 (Hölder mean [27])**.** *The Hölder mean (power mean) of (positive) real numbers x_i, $i = 1, 2, \ldots, n$, is defined as*

$$K(p, 0; \mathbf{x}) \equiv M(p; \mathbf{x}) \triangleq \left(\frac{1}{n}\sum_{i=1}^n x_i^p\right)^{\frac{1}{p}} \tag{7}$$

where $p \in \mathbb{R}$.

Definition 3 (Lehmer mean [27])**.** *The Lehmer mean of (positive) real numbers x_i, $i = 1, 2, \ldots, n$, is defined as*

$$K(p, p-1; \mathbf{x}) \equiv L(p; \mathbf{x}) \triangleq \frac{\sum_{i=1}^n x_i^p}{\sum_{i=1}^n x_i^{p-1}} \tag{8}$$

where $p \in \mathbb{R}$.

Special cases of the Hölder and Lehmer means include[5]:

- Harmonic mean: $H(\mathbf{x}) \triangleq M(-1; \mathbf{x}) \equiv L(0; \mathbf{x})$
- Geometric mean: $G(\mathbf{x}) \triangleq M(0; \mathbf{x})$
- Arithmetic mean: $A(\mathbf{x}) \triangleq M(1; \mathbf{x}) \equiv L(1; \mathbf{x})$
- Quadratic mean: $Q(\mathbf{x}) \triangleq M(2; \mathbf{x})$

It can be further proved [27], [34], [35] that both means are monotonically increasing functions of p, i.e., if $p < q$ then $M(p; \mathbf{x}) < M(q; \mathbf{x})$ as well as $L(p; \mathbf{x}) < L(q; \mathbf{x})$.

Moreover, if $p_1 \le p_2$, $q_1 \le q_2$, then [27]

$$K(p_1, q_1; \mathbf{x}) \le K(p_2, q_2; \mathbf{x}). \tag{9}$$

Thus, Eq. (9) leads to inequalities

$$M(p; \mathbf{x}) \le L(p; \mathbf{x}) \quad \text{if } p \ge 1$$

$$M(p; \mathbf{x}) > L(p; \mathbf{x}) \quad \text{if } p < 1.$$

We now prove the following new upper bound for a general Lehmer mean.

Lemma 5. *Having real positive numbers $x_i > 0$, $i = 1, 2, \ldots, n$, then for a sufficiently large $p \ge p^\star$*

$$L(p; \mathbf{x}) \le L(p-1; \mathbf{x}) \left(1 + \frac{\sum_{i=1}^n x_i^{p-1}\log x_i}{\sum_{i=1}^n x_i^{p-1}} - \frac{\sum_{i=1}^n x_i^{p-2}\log x_i}{\sum_{i=1}^n x_i^{p-2}}\right). \tag{10}$$

[5]The Lehmer mean for $p = 2$, i.e., $L(\mathbf{x}, 2)$ is in literature sometimes called a *contraharmonic* mean.

Remark: If $n=2$, then inequality (10) holds for $p \geq p^\star = 1$, for any values x_i. In general, $p^\star$ depends on the largest inflection point of $L(p;\mathbf{x})$, which, in turn, depends on x_i. However, the inflection points location, or at least bounds on them, are, to the best of our knowledge, unknown [35] and need to be found numerically[6].

Proof. After taking the second derivative, i.e., $\frac{\partial^2 L(p;\mathbf{x})}{\partial p^2}$, and investigating the behaviour at $p \to \infty$ we find that the function $L(p;\mathbf{x})$ for $p \geq p^\star$, where $p^\star$ is the largest inflection point, is a concave function [27], [34]. Therefore, for $p \geq p^\star$,

$$L(p;\mathbf{x}) \leq L(p-1;\mathbf{x}) + \left.\frac{\partial L(q;\mathbf{x})}{\partial q}\right|_{q=p-1}. \qquad (11)$$

Note that for $n=2$, it can be found that $p^\star = 1$ is the only inflection point.

Taking the derivative we obtain (cf. (8))

$$\frac{\partial L(p;\mathbf{x})}{\partial p} = \\ = L(p;\mathbf{x}) \left(\frac{\sum_{i=1}^n x_i^p \log x_i}{\sum_{i=1}^n x_i^p} - \frac{\sum_{i=1}^n x_i^{p-1} \log x_i}{\sum_{i=1}^n x_i^{p-1}} \right). \qquad (12)$$

Plugging (12) into (11) (for $\frac{\partial L(q;\mathbf{x})}{\partial q}|_{q=p-1}$) concludes the proof. ■

B. Relation of Hölder mean and quasi-arithmetic means to consensus algorithms

Comparing the definition of (weighted) Hölder (power) mean (7) [35] and the definition of our state-dependent consensus (1), we observe that if we take the p-th power of the states $x_i(k)$ we obtain

$$x_i^p(k) = \sum_{j \in \mathcal{N}_i^+} w_{ij}(k-1) x_j^p(k-1)$$

and substituing $z_i(k) \equiv x_i^p(k)$, $p \in \mathbb{R}$ yields

$$z_i(k) = \sum_{j \in \mathcal{N}_i^+} w_{ij}(k-1) z_j(k-1),$$

which is a consensus algorithm with states $z_i(k)$, i.e., $\lim_{k\to\infty} z_i(k) = c$, or from global point of view, with $\mathbf{x}^p = (x_1^p, x_2^p, \ldots, x_N^p)$,

$$\lim_{k\to\infty} \mathbf{x}(k) = \left(\prod_{k=0}^{\infty} \mathbf{W}(k) \mathbf{x}^p(0) \right)^{1/p}.$$

For a specific selection of the weights $w_{ij}(k)$ (see Sec. IV ahead) the consensus can be found exactly[7].

In general, the Hölder (power) means are only a special case of the class of so-called *quasi-arithmetic* means [27], i.e. for an arbitrary n,

$$M_\varphi(\mathbf{x}, \boldsymbol{\omega}) = \varphi^{-1} \left(\sum_{i=1}^n w_i \varphi(x_i) \right),$$

with some monotone invertible function $\varphi(\cdot)$, can be viewed, by applying function $\varphi(\cdot)$ on both sides, as a consensus algorithm with states $\varphi(x_i(k))$, i.e.,

$$\varphi(x_i(k)) = \sum_{j \in \mathcal{N}_i^+} w_{ij} \varphi(x_j(k-1)).$$

In general, combining our approach with quasi-arithmetic means, would lead to an update equation of type [36], [37]

$$x_i(k) = \varphi^{-1} \left(\frac{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1)) \varphi(x_j(k-1))}{\sum_{j \in \mathcal{N}_i^+} f_j(x_j(k-1))} \right),$$

where $\varphi(\cdot)$ would be known and applied locally at each node[8].

Clearly, quasi-arithmetic means present an interesting generalization of classical means and may further extend the analysis of the consensus algorithm with state-dependent weights. For example, the bounds on the difference (distance) between two generalized means, i.e., $\rho(M_\varphi(\mathbf{x}, \boldsymbol{\omega}), M_\psi(\mathbf{x}, \boldsymbol{\omega}))$ [38], could not only directly bound the steady state of such algorithm, but can bound the error between the steady states. An interesting question, concerning this generalized approach, is, whether it is possible to find such (network-independent) functions $f(\cdot)$, $\varphi(\cdot)$, for which the steady state would lead to the *sum* of initial values $x_i(0)$, thus bypassing Lemma 1. Such analysis is, however, at the moment, left for future research.

IV. Steady states

In this section we provide examples of weights with specific functions and derive their steady states.

Let us begin with two very simple special cases.

A. Constant initial values

Using Eq. (1) with weights (2), clearly, for any topology if $x_i(0) = x_j(0) = c$, for all i, j, then

$$\lim_{k\to\infty} x_i(k) = x_i(0) = c.$$

B. Fully-connected network

In case of a fully-connected network (network where each node is connected to all other nodes) it is straightforward to find that for any $x_i(0) \in \mathbb{R}$ the steady state is equal to the state after the first iteration, i.e.,

$$\lim_{k\to\infty} x_i(k) = x_i(1) = \frac{\sum_{j=1}^N f_j(x_j(0);0) x_j(0)}{\sum_{j=1}^N f_j(x_j(0);0)}.$$

C. Steady states –special cases

First, let us recall the properties of the weight consensus matrix as proposed in [2], [39].

Lemma 6. *Having a static connected network described by an adjacency matrix* $\mathbf{A}$ *with a degree matrix* $\mathbf{D} = \operatorname{diag}(d_1, d_2, \ldots, d_N)$*, then the weight matrix*

$$\mathbf{W} = (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}) \qquad (13)$$

[6]Nevertheless, based on the vast number of our simulations, the largest inflection point $p^\star$ is less than 2 almost surely.

[7]In case of standard, well-known, weights which guarantee convergence to the average, e.g., Metropolis weights [3], the algorithm converges to a general Hölder (power) mean of the initial values.

[8]In our case, the function $\varphi(x_i) = x_i$.

has the following properties:

1) $\mathbf{W}\mathbf{1} = \mathbf{1}$,
2) *maximum eigenvalue* $\lambda_{\max} \equiv \max_i |\lambda_i| = 1$, *with corresponding right eigenvector* $\mathbf{v}_{\max} = \mathbf{1}$,
3) *left eigenvector corresponding to* $\lambda_{\max}$, $\mathbf{u}_{\max}^\top = \frac{1}{\sum_{i=1}^N 1+d_i}(1+d_1, 1+d_2, \ldots, 1+d_N)$.

Thus,

$$\lim_{k\to\infty} \mathbf{W}^k = \mathbf{v}_{\max}\mathbf{u}_{\max}^\top.$$

Proof. Property 1) is straighforward to show just by multiplying $\mathbf{W}$ with $\mathbf{1}$ from right. Since $\mathbf{W}$ is a primitive nonnegative matrix Property 2) is a consequence of Perron-Frobenius theorem [2]. Finally, property 3) can be proved by simple multiplication $\mathbf{W}$ with $\mathbf{u}$ from left. ∎

Note that a consensus algorithm with weights (13) does not, in general, lead to an *average* consensus algorithm [2], unless the network has a regular topology, or a combination of two algorithms is performed[9] [5]. Also note, that in this case, the decomposition (3), takes the form, $\mathbf{D}_1(k) = \mathrm{diag}(\frac{1}{1+d_i}, \ldots,)$, $\mathbf{D}_2(k) = \mathbf{I}$, which is the case of the following theorem.

Theorem 3. *Assuming a static connected network, then for any initial number* $x_i(0) \in \mathbb{R}$ *(*$\forall i = 1, 2, \ldots, N$*), the update algorithm* (1) *with weight functions (cf. Eq.* (2)*)*

$$f_i(x_i(k)) = 1, \quad i = 1, 2, \ldots, N, \tag{14}$$

asymptotically converges to the consensus

$$\lim_{k\to\infty} x_i(k) = \sum_{i=1}^N \frac{1+d_i}{\sum_{j=1}^N (1+d_j)} x_i(0).$$

Note that weights (14) are simply selected only according to the *number* of received messages at time k. A deeper analysis of this algorithm, including its convergence rate, can be found in [5].

Proof. The convergence of the algorithm to *a* consensus follows from Theorem 1. Moreover, from a global point of view,

$$\begin{aligned}\mathbf{x}(k) &= \mathbf{W}(k-1)\mathbf{x}(k-1) = (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\mathbf{x}(k-1)\\ &= \left((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\right)^k \mathbf{x}(0).\end{aligned}$$

Taking the results of Lemma 6 concludes the proof. ∎

Conjecture 1. *Theorem 1 holds also for (appropriate) mixed positive/negative functions* $f_i(\cdot\,; k)$*, as long as the weights at each node sum to* 1 *(and having non-zero denominator in* (2)*), or for specific topologies.*

Remark: If $f_i(x_i(k)) < 0$ for all $i \in \{1, 2, \ldots, N\}$ and k, then $0 < w_{ij} < 1$, $\forall i, j$, and the algorithm is equivalent to case of $f_i(x_i(k)) > 0$ for all i, k (see the Case 1 of the proof of Theorem 1 for strictly positive case). For a more general (mixed) case, see Theorem 4 ahead.

To support Conjecture 1, we define the following algorithm, which can take also negative values, but which, nevertheless, converges to a consensus.

[9] Unlike the case considered here, in [5] the network is considered to be without self-loops, i.e., $\tilde{\mathbf{W}} = \mathbf{D}^{-1}\mathbf{A}$. Nevertheless, our approach with weight matrix (13) is equivalent to that of [5], as shown in [2].

Theorem 4. *For any number* $x_i(0) \in \mathbb{R}\backslash\{0\}$ *(*$i = 1, 2, \ldots, N$*), the algorithm* (1) *with weight functions (cf. Eq.* (2)*)*

$$f_i(x_i(k)) = \frac{1}{x_i(k)} \tag{15}$$

asymptotically converges to the consensus, i.e.,

$$\lim_{k\to\infty} x_i(k) = \frac{\sum_{i=1}^N 1+d_i}{\sum_{i=1}^N \frac{1+d_i}{x_i(0)}} \quad, \forall i = 1, 2, \ldots, N.$$

Corollary 2. *In case of a regular network (same degree* d_i *of every node),*

$$\lim_{k\to\infty} x_i(k) = \frac{N}{\sum_{i=1}^N \frac{1}{x_i(0)}},$$

thus, the algorithm from Theorem (4) *with weight functions* (15) *converges to a steady state equal to the harmonic mean of the initial values.*

Proof of Theorem 4. By plugging the weights (15) into (2), Eq. (1) yields

$$x_i(k) = \frac{1}{\frac{1}{x_i(k-1)} + \sum_{j'\in\mathcal{N}_i} \frac{1}{x_{j'}(k-1)}} + \sum_{j\in\mathcal{N}_i} \frac{1}{\frac{1}{x_i(k-1)} + \sum_{j'\in\mathcal{N}_i} \frac{1}{x_{j'}(k-1)}},$$

which can be rearranged as

$$\frac{1}{x_i(k-1)} + \sum_{j'\in\mathcal{N}_i} \frac{1}{x_{j'}(k-1)} = \frac{1+d_i}{x_i(k)}.$$

From a global (network) point of view, we can write

$$(\mathbf{I}+\mathbf{A})\hat{\mathbf{x}}(k-1) = (\mathbf{I}+\mathbf{D})\hat{\mathbf{x}}(k),$$

where $\hat{\mathbf{x}}(k) = (\frac{1}{x_1(k)}, \frac{1}{x_2(k)}, \ldots, \frac{1}{x_N(k)})^\top$.

Thus, we obtain

$$\begin{aligned}\hat{\mathbf{x}}(k) &= (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\hat{\mathbf{x}}(k-1),\\ &= \left((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\right)^k \hat{\mathbf{x}}(0)\end{aligned}$$

and using Lemma 6 we find

$$\lim_{k\to\infty} \mathbf{x}(k) = \frac{\sum_{i=1}^N 1+d_i}{\sum_{i=1}^N \frac{1+d_i}{x_i(0)}}\mathbf{1}.$$

Corollary 2 follows, if $d_i = d, \forall i$. ∎

Let us now analyze functions for which the exact steady state may not be found analytically.

Theorem 5. *Assuming a static connected network, then for any non-negative number* $x_i(0) \geq 0$ *(*$i = 1, 2, \ldots, N$*), the algorithm* (1) *with weight functions (cf. Eq. 2)*

$$f_i(x_i(k)) = x_i(k) \tag{16}$$

asymptotically converges to a consensus, i.e.,

$$\lim_{k\to\infty} x_i(k) = c^{(x)} \quad, \forall i = 1, 2, \ldots, N.$$

and is bounded

$$\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0)\le c^{(x)}\le$$
$$\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0)+\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0)\log x_i(0)-$$
$$-\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0)\cdot\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}\log x_i(0).$$

Corollary 3. *In case of a regular network (same degree d of every node),*

$$\frac{1}{N}\sum_{i=1}^{N}x_i(0)\le c^{(x)}\le$$
$$\frac{1}{N}\sum_{i=1}^{N}x_i(0)+\frac{1}{N}\sum_{i=1}^{N}x_i(0)\log x_i(0)-$$
$$-\frac{1}{N}\sum_{i=1}^{N}x_i(0)\cdot\frac{1}{N}\sum_{i=1}^{N}\log x_i(0),$$

thus, the algorithm converges to a steady state bigger than the average of the initial values.

Proof of Theorem 5: Convergence to a consensus follows from Theorem 1. The lower bound follows from the general inequality of Lehmer means (9), i.e., $L(1;\mathbf{x})\le L(2;\mathbf{x})$. Since $\frac{1}{1+d_i}\sum_{j\in\mathcal{N}_i^+}x_i(k)\le\frac{\sum_{j\in\mathcal{N}_i^+}x_i^2(k)}{\sum_{j\in\mathcal{N}_i^+}x_i(k)}$, or globally (cf. Theorem 3)

$$((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}))^k\mathbf{x}(0)\le\frac{\sum_{j\in\mathcal{N}_i^+}x_j^2(k)}{\sum_{j\in\mathcal{N}_i^+}x_j(k)}\quad(\forall i),$$

as $k\to\infty$ we obtain the lower bound.

As stated in the Remark in Sec. III-A, the inflection point $p^\star$ of Lehmer mean is typically less than 2. Thus, from Eq. (10) for $p=2$, we may write for the upper bound

$$\frac{\sum_{j\in\mathcal{N}_i^+}x_j^2(k)}{\sum_{j\in\mathcal{N}_i^+}x_j(k)}\le\frac{1}{1+d_i}\sum_{j\in\mathcal{N}_i^+}x_j(k)+$$
$$+\frac{1}{1+d_i}\sum_{j\in\mathcal{N}_i^+}x_j(k)\log x_j(k)-$$
$$-\frac{1}{1+d_i}\sum_{j\in\mathcal{N}_i^+}x_j(k)\frac{1}{1+d_i}\sum_{j\in\mathcal{N}_i^+}\log x_j(k).\quad(17)$$

The terms of the upper bound can be written globally as

$$(\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\mathbf{x}(k)+(\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\mathbf{z}(k)-$$
$$\big((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\mathbf{x}(k)\big)\circ\big((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})\log\mathbf{x}(k)\big)=$$
$$=((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}))^k\mathbf{x}(0)+((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}))^k\mathbf{z}(0)-$$
$$\big(((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}))^k\mathbf{x}(0)\big)\circ\big(((\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}))^k\log\mathbf{x}(0)\big).$$

where $\mathbf{z}(k)=\mathbf{x}(k)\circ\log\mathbf{x}(k)$.

Taking $k\to\infty$ concludes the proof. ∎

It must be noted that although according to our simulations the largest inflection point of Lehmer mean (see Definition 3) is always smaller than 2, a rigorous proof for this claim is missing. Nonetheless, in case inequality (17) does not hold, i.e., $p^\star>2$, we can still upper bound the steady state $c^{(x)}$ by $x_{\max}(0)$ (see Lemma 1).

D. Bounds on more complex functions

Naturally, it is of interest to find bounds for more complex functions. A possible approach is to use Theorem 2 and some approximation of the function.

As an example, consider the following theorem.

Theorem 6. *Assuming a static connected network, then for any non-negative number $x_i(0)\ge 0$ ($i=1,2,\dots,N$), algorithm* (1) *with weight functions (cf. Eq. 2)*

$$f_i(x_i(k))=\exp(x_i(k))\quad(18)$$

asymptotically converges to a consensus, i.e.,

$$\lim_{k\to\infty}x_i(k)=c^{(\exp)}\quad,\forall i=1,2,\dots,N.$$

and the states $x_i(k)$ are bounded from below

$$\frac{\sum_{j\in\mathcal{N}_i^+}(1+x_i(k))x_i(k)}{\sum_{j\in\mathcal{N}_i^+}(1+x_i(k))}\le\frac{\sum_{j\in\mathcal{N}_i^+}\exp(x_i(k))x_i(k)}{\sum_{j\in\mathcal{N}_i^+}\exp(x_i(k))}.$$

Furthermore, if $x_i(0)<1$, then the states are upper-bounded

$$\frac{\sum_{j\in\mathcal{N}_i^+}\exp(x_i(k))x_i(k)}{\sum_{j\in\mathcal{N}_i^+}\exp(x_i(k))}\le\frac{\sum_{j\in\mathcal{N}_i^+}x_i^2(k)}{\sum_{j\in\mathcal{N}_i^+}x_i(k)}.\quad(19)$$

Proof. Taking the Taylor expansion we obtain $\exp(x)\approx 1+x+\dots$, and thus for $x\ge 0$, it holds that $1+x\le\exp(x)$. We now check if the conditions of Theorem 2 are satisfied. By taking the derivative we find that $\frac{1+x}{\exp(x)}$ is a decreasing function. Furthermore, $\frac{\mathrm{d}}{\mathrm{d}x}(1+x)\le 1+x$ for $x\ge 0$, and thus the lower bound yields

$$\frac{\sum_{j\in\mathcal{N}_i^+}(1+x_j(k))x_j(k)}{\sum_{j\in\mathcal{N}_i^+}(1+x_j(k))}\le\frac{\sum_{j\in\mathcal{N}_i^+}\exp(x_j(k))x_j(k)}{\sum_{j\in\mathcal{N}_i^+}\exp(x_j(k))}.$$

Similarly, if $x_i(k)<1$, $\frac{\exp(x)}{x}$ satisfies the sufficient conditions, and the upper bound follows straightforwardly from Theorem 2. ∎

Theorem 6 thus states that for any k the consensus algorithm with weight functions $f(x_i)=\exp(x_i)$ is bounded from below by the consensus algorithm with weight functions $f(x_i)=1+x_i$. Similarly, from above by the consensus algorithm with weight functions $f(x_i)=x_i$. The steady state is then bounded by the following corollary.

Corollary 4. *If $x_i(0)<1$, then the steady state $c^{(\exp)}$ of the consensus algorithm with weight function $f(x)=\exp(x)$ from*

Theorem 6 is bounded as

$$\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0) < c^{(\exp)} <$$
$$< \sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0) + \sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0)\log x_i(0) -$$
$$- \sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}x_i(0)\cdot\sum_{i=1}^{N}\frac{1+d_i}{\sum_j(1+d_j)}\log x_i(0). \quad (20)$$

Proof. Using Theorem 2 we observe that $\frac{\sum_{j\in\mathcal{N}_i^+}x_j(k)}{\sum_{j\in\mathcal{N}_i^+}1} < \frac{\sum_{j\in\mathcal{N}_i^+}(1+x_j(k))x_j(k)}{\sum_{j\in\mathcal{N}_i^+}(1+x_j(k))}$. The lower bound then follows from Theorem 3. An alternative bound can be found using Eq. (5) and the bound on steady state of $x_i(k) \leq \frac{1}{1+d_i}\sum_{j\in\mathcal{N}_i^+}x_j^2(k-1)$. However, this bound is in case of $x_i(0)<1$ looser.

The upper bound follows from the upper bound from Theorem 5. ■

Note that by finding better approximations and functions for bounding one function by another may lead to even tighter bounds. Nevertheless, as we will show later in the simulation section, the derived bounds can be very tight, and thus the steady states of such complicated functions can be estimated quite accurately. Although a second-order analysis of these bounds may be interesting, such analysis seems to be complicated and is therefore omitted here.

V. Convergence rate

It is known that the convergence rate of the product of row-stochastic matrices, which converge to a "consensus" matrix, is at least geometric [15], [20], [40], [41].

It is also well-known that in case of consensus algorithms with static non-varying network, i.e., consensus of type

$$\mathbf{x}(k) = \mathbf{W}\mathbf{x}(k-1),$$

the convergence rate depends on the second largest eigenvalue of the weight matrix $\mathbf{W}$. A special case of this type, with weight matrix $\mathbf{W} = (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A})$, are algorithms from Theorem 3 and Theorem 4, i.e., consensus algorithms with state functions $f(x_i(k)) = 1$ and $f(x_i(k)) = \frac{1}{x_i(k)}$ (see the proofs of the theorems). The convergence rate of such algorithm is discussed in [5].

Moreover, as stated in Sec. II, in case the weight functions are node and time-independent, i.e., $f(x_i(k))$, the weight matrix $\mathbf{W}(k)$ converges to a fixed known consensus matrix, i.e.,

$$\lim_{k\to\infty}\mathbf{W}(k) = (\mathbf{I}+\mathbf{D})^{-1}(\mathbf{I}+\mathbf{A}). \quad (21)$$

Thus, eventually, the convergence rate is independent from the function $f(x_i(k)$ and is solely dependent only the network topology (see Fig. 4 ahead).

Naturally, in the transient phase the weight functions may influence the convergence rate. Since

$$\text{trace}\left(\mathbf{W}(k)\right) = \sum_{i=1}^{N}\frac{f_i(x_i(k);k)}{\sum_{j\in\mathcal{N}_i^+}f_j(x_j(k);k)} = \sum_{i=1}^{N}\lambda_i(k),$$

where $1=\lambda_1(k) > \lambda_2(k) \geq \ldots \lambda_N(k)$ are the eigenvalues of $\mathbf{W}(k)$, or more specifically

$$\lambda_2(k) = \sum_{i=1}^{N}\frac{f_i(x_i(k);k)}{\sum_{j\in\mathcal{N}_i^+}f_j(x_j(k);k)} - 1 - \sum_{i=3}^{N}\lambda_i(k).$$

Thus, by modifying $f_i(k)$ we may also modify $\lambda_2(k)$.

There exist many bounds on λ_2 in literature, e.g., [42]–[44]. We here propose a bound on $\lambda_2(k)$ for our algorithm. This bound is based on the bound from [44].

Theorem 7. *The second largest eigenvalue of $\mathbf{W}(k)$ with weights (2) is bounded as*

$$\lambda_2(k) \leq \max\{A_1, A_2\},$$

where

$$A_1 = \sqrt{\frac{N-2}{N-1}\left(Q - \frac{Q^2}{N-1}\right)} - \frac{Q}{N-1},$$
$$A_2 = \frac{Q}{N-1} + \sqrt{\frac{N-2}{N-1}\left(Q - \frac{Q^2}{N-1}\right)},$$

with $Q = \sum_{i=1}^{N}\frac{f_i(x_i(k);k)}{\sum_{j\in\mathcal{N}_i^+}f_j(x_j(k);k)} - 1$.

Proof. Using the bounds from [44, Theorem 21] for a matrix $\mathbf{A}$ and having $\lambda_1 = 1$, i.e.,

$$c_1 = \frac{\text{trace}(\mathbf{A})-1}{N-1} - \sqrt{\frac{N-2}{N-1}q(\mathbf{A})},$$
$$d_1 = \frac{\text{trace}(\mathbf{A})-1}{N-1} + \sqrt{\frac{N-2}{N-1}q(\mathbf{A})},$$

with $q(\mathbf{A}) = \text{trace}\left(\left(\mathbf{A} - \frac{\text{trace}(\mathbf{A})}{N}\mathbf{I}\right)^2\right) - \frac{N}{N-1}\left(1 - \frac{\text{trace}(\mathbf{A})}{N}\right)^2$, then, by plugging in $\mathbf{W}(k)$ we find that

$$q(\mathbf{W}(k)) = \sum_{i=2}^{N}\lambda_i^2 - \frac{(\sum_{i=2}^{N}\lambda_i)^2}{N-1}.$$

Due to the fact[10] that $\text{trace}\left(\mathbf{W}(k)^2\right) \leq \text{trace}\left(\mathbf{W}(k)\right)$ and $\lambda_1 = 1$, it follows that

$$q(\mathbf{W}(k)) = \sum_{i=2}^{N}\lambda_i^2 - \frac{(\sum_{i=2}^{N}\lambda_i)^2}{N-1} \leq \sum_{i=2}^{N}\lambda_i - \frac{(\sum_{i=2}^{N}\lambda_i)^2}{N-1}.$$

Plugging in the equality $\sum_{i=2}^{N}\lambda_i(k) = \text{trace}\left(\mathbf{W}(k)\right) - 1 = \sum_{i=1}^{N}\frac{f_i(x_i(k);k)}{\sum_{j\mathcal{N}_i^+}f_j(x_j(k);k)} - 1$ we obtain the bounds. ■

By proper selection of weight functions we may thus influence the convergence rate (especially in the transient phase) of the algorithm. Although the bound provided here may be very loose [44], it is easy to compute in case of our algorithm. We must note here that, in general, the second largest eigenvalue $\lambda_2(k)$ may not be the best convergence

[10] Since $\text{trace}\left(\mathbf{W}(k)^2\right) = \sum_{i=1}^{N}\Big(\frac{f_i(\cdot)}{\sum_{j\in\mathcal{N}_i^+}f_j(\cdot)}\cdot\sum_{j'\in\mathcal{N}_i^+}\frac{f_{j'}(\cdot)}{\sum_{j''\in\mathcal{N}_{j'}^+}f_{j''}(\cdot)}\Big)$, then by factorizing we find that for $f_i(\cdot)\geq 0$, $\Big(\sum_{j'\in\mathcal{N}_i^+}\frac{f_{j'}(\cdot)}{\sum_{j''\in\mathcal{N}_{j'}^+}f_{j''}(\cdot)}\Big) \leq 1$ for all i. Therefore $\text{trace}\left(\mathbf{W}(k)^2\right) \leq \text{trace}\left(\mathbf{W}(k)\right)$.

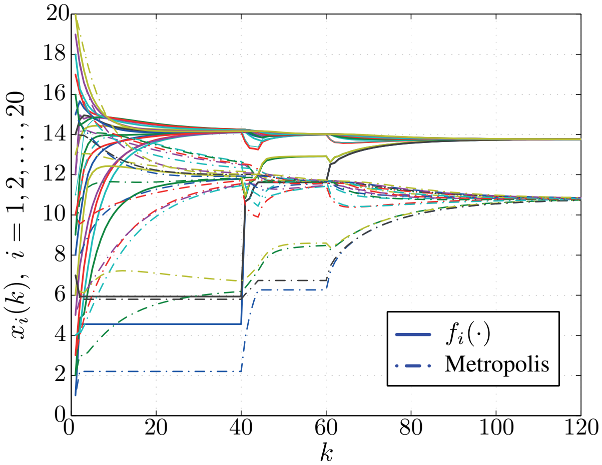

Fig. 1. Example of Alg. (1) with weights as $f_i(x_i(k)) = x_i(k)$ and two disconnected nodes during a finite number of iterations. Dash-dotted lines are Metropolis weights [3].

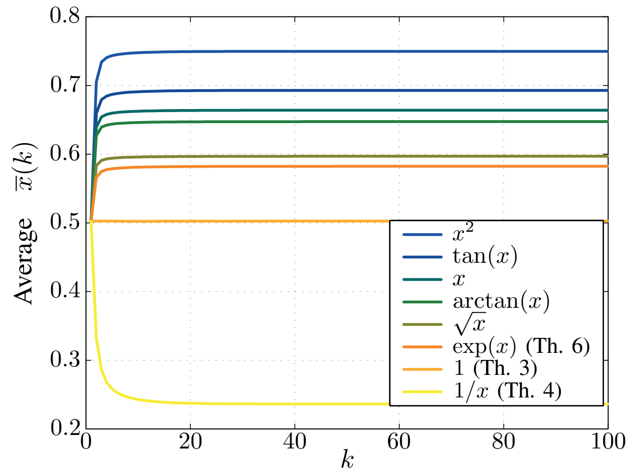

Fig. 2. Various weight functions. Averaged over 100 random initializations and over 20 nodes; $x_i(0) \in (0,1)$.

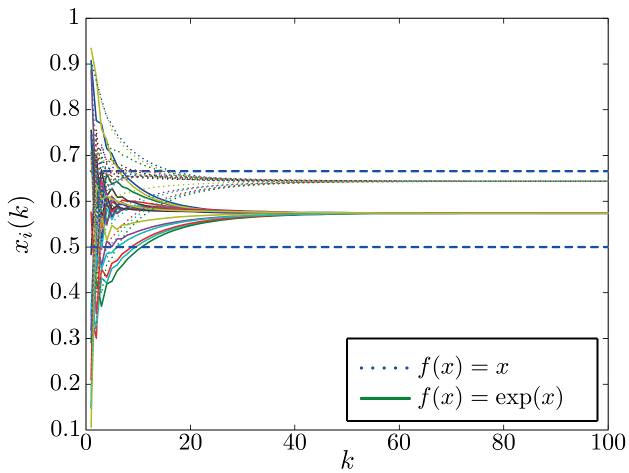

Fig. 3. Behaviour of state-dependent consensus with weight functions (16) and (18). Observe that the states $x_i(k)$ in case of weight functions $f(x_i(k)) = x_i(k)$ are always above the states $x_i(k)$ in case the weight functions are $f(x_i(k)) = \exp(x_i(k))$ (Eq. (19)). Dashed lines are the bounds on steady states, Eq. (20).

rate measure in case of time-varying matrices. Coefficients of ergodicity (see proof of Theorem 1) or other measures with multiplicative property[11] may possibly be more appropriate and easier to analyze. However, we leave this question for future research.

VI. Simulations

We simulate random geometric networks (network with nodes communicating only with neighbors within some radius) with number of nodes $N = 20$.

In Fig. 1 we show an example of Theorem 1 for the weight functions $f_i(x_i(k)) = x_i(k)$, with initialization $x_i(0) = \{1, 2, \ldots, 20\}$, thus $\bar{x}(0) = 10.5$. We simulate the case that the two nodes are disconnected for iterations $k_0 = \{3, 4, \ldots, 40\} \cup \{45, 46, \ldots, 60\}$, i.e., $f_j(x_j(k_0)) = 0, \forall j \in \{\mathcal{N}_1^+ \cup \mathcal{N}_7^+\}$. We observe that the algorithm still reaches a consensus as expected from Theorem 1. We compare these weights with so-called Metropolis weights [3] (dash-dotted lines) whose weights require the knowledge of the node degrees in the network. We observe that in that case the convergence is slightly slower ($k = 120$ vs. $k = 100$, cf. Fig. 4), and due to the disconnected nodes, the states also do not converge to the average of the initial values ($\bar{x}(k) \approx 10.7$). Also we can see that the steady state of the consensus algorithm with weight function $f(x_i(k)) = x_i(k)$ is above the average of initial values (steady state in case of Metropolis weights) as expected by Theorem 5.

In Fig. 2 we simulate eight different weights with following functions: $f_i^{(1)}(x_i(k)) = x_i^2(k)$, $f_i^{(2)}(x_i(k)) = \tan(x_i(k))$, $f_i^{(3)}(x_i(k)) = x_i(k)$, $f_i^{(4)}(x_i(k)) = \arctan(x_i(k))$, $f_i^{(5)}(x_i(k)) = \sqrt{x_i(k)}$, $f_i^{(6)}(x_i(k)) = \exp(x)$, $f_i^{(7)}(x_i(k)) = 1$, and $f_i^{(8)}(x_i(k)) = \frac{1}{x_i(k)}$. The simulations have been performed for 100 random initializations $x_i(0) \in (0,1)$, $i = 1, 2, \ldots, N$, for a fixed randomly selected geometric network. The depicted results are the averaged values over the initializations and nodes. We observe that in all cases the states converge to a consensus as expected.

In Tab. I and Fig. 3 we investigate the tightness of the bounds of Theorem 5 and Theorem 6. We depict the average relative error of the bounds over 1 000 initialization in one random network with $N = 20$, i.e., $\bar{e}_1 = \frac{1}{1000} \sum_{l=1}^{1000} \frac{|\tilde{c}_l - c_l|}{c_l}$, where $\tilde{c}$ is the upper/lower bound on the real steady state c. Similarly, for the relative error between the true steady states $c^{(x)}$ from Theorem 5 and the true steady state $c^{(\exp)}$ from Theorem 6, we obtain $\bar{e}_2 = \frac{1}{1000} \sum_{l=1}^{1000} \frac{|c_l^{(x)} - c_l^{(\exp)}|}{c_l^{(\exp)}} = 0.144$. Thus, as expected from Eq. (19), the lower bound on steady state in case $f(x_i(k)) = x_i(k)$ (Theorem 5) by the steady

weight function $f(x)$	average relative error of lower bound	average relative error of upper bound
x	0.242	0.1166
$\exp(x)$	0.136	0.2797

TABLE I
Tightness of the bounds (20).

[11] Multiplicative property of some matrix measure $\mu(\mathbf{A})$ means that the product of matrices corresponds to a product of measures, i.e., $\mu(\mathbf{A}_1\mathbf{A}_2 \ldots) = \mu(\mathbf{A}_1)\mu(\mathbf{A}_2) \ldots$.

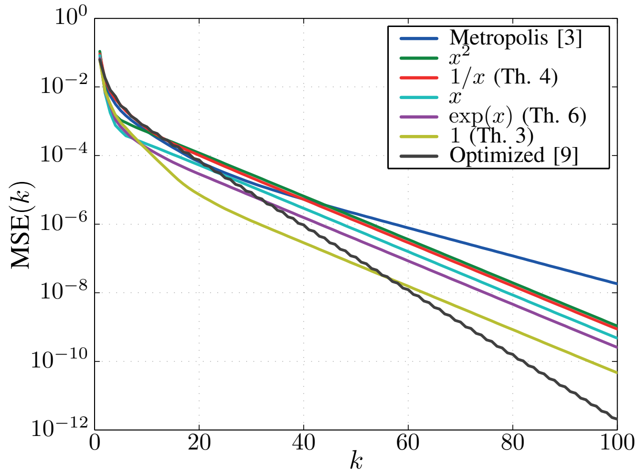

Fig. 4. Convergence rate for various weights. Metropolis weights [3] are slowest, optimized weights [9] are fastest. As k grows, the convergence rate (slope) depends less and less on the function used.

state in case of $f(x_i(k)) = \exp(x_i(k))$ (Eq. (19)) is tighter than the lower bound based on $f(x_i(k)) = 1$ (Theorem 3).

In Fig. 4 we depict the convergence rate for several types of weights in terms of $\mathrm{MSE}(k) = \frac{1}{20}\sum_{i=1}^{20}(x_i^{(\text{type})}(k) - c^{(\text{type})})^2$, where $c^{(\text{type})}$ is the true steady state for the simulation run. From the slopes of the curves we observe that the slowest convergence rate is in case of Metropolis weights [3]. On the other side, the convex optimized weights [9] are the fastest. As we discussed in Sec. V, if the weight function $f_i(\cdot)$ does not depend on the node and time, but depends only on the received state, i.e., $f(\cdot)$, the weights eventually converge to constant weights (21) (In Fig. 4 the lines are parallel). However, we also observe that in the transient phase (up to $k = 20$), the convergence rate depends on the weight function and may be faster than in the case of the convex optimized weights (compare weights with $f(x_i(k)) = 1$ and optimized weights in Fig. 4). Note that although Metropolis and optimized weights guarantee convergence to the average of initial values, whereas other weights do not, they require to have some non-local knowledge of the network.

VII. Conclusion

We provided a novel approach for designing weights for distributed consensus algorithms, without *any* knowledge about the network, based only on the received data. We analyzed this approach and show the properties of the weights in terms of the weight functions. We derived steady states and bounds on the steady states for specific weight functions. Furthermore, we showed how the weight functions influence the convergence rate. A generalized proof for negative weights, as proposed by the conjecture, remains an open issue. The converse problem, i.e., to find appropriate functions for a desired steady state remains a challenging open question for future research[12]. The selection of functions which would optimize the convergence rate (especially in the transient phase) and still converge to a steady state in desired bounds is left open as well.

An interesting extension of our approach would be a consensus algorithm with multidimensional states (matrices). The weight functions, in this case, could be matrix functions, e.g., $F_i(\mathbf{X}_i(k)) = \mathrm{trace}\,(\mathbf{X}_i(k))$ [11]. Further investigation of this approach is left for future research as well.

[12]Such problem is yet little studied in the theory of nonhomogeneous Markov chains as well [45].

References

[1] O. Slučiak and M. Rupp, "Consensus algorithms with state-dependent weights," in *Proc. of the 39th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Florence, Italy, 2014, pp. 5499–5503.

[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, Jan. 2007.

[3] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. ACM/IEEE IPSN–05*, Los Angeles, USA, Apr. 2005, pp. 63–70.

[4] V. Schwarz and G. Matz, "Mean-square optimal weight design for average consensus," in *Proc. IEEE SPAWC-2012*, Jun. 2012, vol. 1, pp. 374–378.

[5] A. Olshevsky and J. N. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Review*, vol. 53, no. 4, pp. 747–772, 2011.

[6] V. Schwarz and G. Matz, "Nonlinear average consensus based on weight morphing," in *Proc. of the 36th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2012, pp. 3129–3132.

[7] L. Georgopoulos and M. Hasler, "Nonlinear Average Consensus," in *International Symposium on Nonlinear Theory and its Applications*, Sapporo, Japan, Oct. 2009.

[8] O. Slučiak and M. Rupp, "Almost sure convergence of consensus algorithms by relaxed projection mappings," in *Proc. of Statistical Sig. Proc. Workshop (SSP)*, Ann Arbor, MI, USA, Aug. 2012, pp. 632–635.

[9] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, pp. 65–78, 2004.

[10] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel Type of Phase Transition in a System of Self-Driven Particles," *Phys. Rev. Lett.*, vol. 75, no. 6, pp. 1226–1229, 1995.

[11] O. Hlinka, O. Slučiak, F. Hlawatsch, and M. Rupp, "Distributed data fusion using iterative covariance intersection," in *Proc. of the 39th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 1880–1884.

[12] O. Slučiak and M. Rupp, "Reaching consensus in asynchronous WSNs: Algebraic approach," in *Proc. of the 36th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, Czech Rep., May 2011, pp. 3300–3303.

[13] Q. Jiang, "Construction of Transition Matrices of Reversible Markov Chains," M.S. thesis, University of Windsor, Ontario, Canada, 2009.

[14] F. P. Kelly, *Reversibility and Stochastic Networks*, Cambridge Mathematical Library, 2011.

[15] S. Chatterjee and E. Seneta, "Towards Consensus: Some Convergence Theorems on Repeated Averaging," *Journal of Applied Probability*, vol. 14, pp. 89–97, 1977.

[16] B. Touri, *Product of Random Stochastic Matrices and Distributed Averaging*, Ph.D. thesis, University of Illinois at Urbana-Champaign, 2012.

[17] J. Hajnal and M. S. Bartlett, "The ergodic properties of nonhomogeneous finite Markov chains," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 52, pp. 67–77, 1956.

[18] J. Hajnal, "Weak ergodicity in non-homogeneous Markov chains," *Proc. Camb. Phil. Soc.*, vol. 54, pp. 233–246, 1958.

[19] J. Shen, "A geometric approach to ergodic non-homogeneous Markov chains," in *Wavelet Analysis and Multiresolution Methods*, 2000, vol. 212 of *Lecture Notes in Pure and Appl. Math.*, pp. 341–366.

[20] E. Seneta, *Non-negative Matrices and Markov Chains*, Springer, 2006.

[21] A. E. Brouwer and W. H. Haemers, *Spectra of graphs*, Springer, 2012.

[22] J. Wolfowitz, "Products of indecomposable, aperiodic, stochastic matrices," *Proc. of the American Mathematical Society*, 1963.

[23] T. A. Sarymsakov, "On the ergodic principle for non-stationary Markov chains," *Dokl. Akad. Nauk. SSSR (N.S.)*, vol. 90, 1953.

[24] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," *IEEE Conf. on Decision and Control*, pp. 2996–3000, 2005.

[25] J. L. Mott, “Conditions for the Ergodicity of Non-homogeneous Finite Markov Chains,” *Proc. Roy. Soc. Edinburgh*, vol. 64, pp. 369–380, 1957.
[26] A. Leizarowitz, “On Infinite Products of Stochastic Matrices,” *Linear Algebra and Its Applications*, , no. 168, pp. 189–219, 1992.
[27] P. S. Bullen, *Handbook of Means and Their Inequalities (2nd Ed.)*, Mathematics and Its Applications. Springer, 2003.
[28] Z. Daróczy, “Einige Ungleichungen über die mit Gewichtsfunktionen gebildeten Mittelwerte,” *Monatshefte für Mathematik*, vol. 68, pp. 102–112, 1964.
[29] L. Losonczi, “Über eine neue Klasse von Mittelwerten,” *Acta Sci. Math. (Szeged)*, vol. 32, no. 1–2, pp. 71–81, 1971.
[30] Z. Daróczy and L. Losonczi, “Über den Vergleich von Mittelwerten,” *Publ. Math. Debrecen*, vol. 17, pp. 289–297, 1970.
[31] R. Mesiar and J. Špirková, “Weighted means and weighting functions,” *Kybernetika*, vol. 42, no. 2, pp. 151–160, 2006.
[32] J. Špirková and P. Král’, “Mixture operators and the additional insurance,” in *Proceedings of the 5th EUSFLAT Conference*, 2007, vol. 2.
[33] D. Farnsworth and R. Orr, “Gini Means,” *The American Mathematical Monthly*, vol. 93, no. 8, pp. 603–607, 1986.
[34] S. Sýkora, “Mathematical Means and Averages: Basic Properties,” *Ed. S. Sykora*, vol. III, 2009, [Online] http://dx.doi.org/10.3247/SL3Math09.001.
[35] O. Slučiak and M. Rupp, “On the maximum number of inflection points of the power mean function,” *Journal of Mathematical Inequalities*, 2014, (submitted) http://publik.tuwien.ac.at/files/PubDat_229829.pdf.
[36] J. Špirková, “Equality of special mixture operators and quasi-arithmetic means,” *Acta Univ. M. Belii, ser. Math.*, vol. 14, pp. 61–71, 2007.
[37] L. Losonczi, “Equality of two variable weighted means: Reduction to differential equations,” *Aequationes Mathematicae*, vol. 58, pp. 223–241, 1999.
[38] P. Pasteczka, “A new estimate of the difference among quasi-arithmetic means,” *arXiv:1310.7212v1*, 2013.
[39] A. Jadbabaie, Jie Lin, and A.S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Trans. on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.
[40] J. M. Anthonisse and H. Tijms, “Exponential Convergence of Products of Stochastic Matrices,” *Journal of Mathematical Analysis and Applications*, vol. 59, pp. 360–364, 1977.
[41] A. Federgruen, “The rate of convergence for backwards products of a convergent sequence of finite Markov matrices,” *Stochastic Processes and their Applications*, vol. 11, pp. 187–192, 1981.
[42] H. J. Landau and A. M. Odlyzko, “Bounds for Elgenvalues of Certain Stochastic Matrlces,” *Linear Algebra and Its Applications*, vol. 38, pp. 5–15, 1981.
[43] U. G. Rothblum and C. P. Tan, “Upper Bounds on the Maximum Modulus of Subdominant Eigenvalues of Nonnegative Matrices,” *Linear Algebra and its Applications*, vol. 66, pp. 45–86, 1985.
[44] O. Rojo, R. Soto, and H. Rojo, “Bounds for Sums of Eigenvalues and Applications,” *Computers and Mathematics with Applications*, vol. 39, pp. 1–15, 2000.
[45] D. Teodorescu, “Nonhomogeneous Markov chains with desired properties – the nearest Markovian model,” *Stochastic Processes and their Applications*, vol. 10, pp. 255–270, 1980.