Gossip Algorithms for Computing U-statistics

Kristiaan Pelckmans * **Johan A.K. Suykens** **

* *Division of Systems and Control Department of Information Technology, Uppsala University, SE-751 05 Uppsala, Sweden (kp@it.uu.se)*
** *KULeuven - ESAT - SCD/SISTA, 3001 Leuven, Belgium, (johan.suykens@esat.kuleuven.be)*

Abstract: This manuscript studies two 'randomized gossip' algorithms for computing a U-statistic over a weighted, (un)directed graph. We propose algorithms $U1$-*gossip* and $U2$-*gossip* and derive exponential convergence to a global consensus. The proofs rely on a convergence result for Markov chains and on a probabilistic concentration inequality.

Keywords: Gossip algorithm, U-statistics, Markov chains, Concentration inequality.

1. INTRODUCTION

A *gossip algorithm* is a decentralized algorithm which computes a global quantity (or an *estimate*) by repeated application of a local computation, following the topology of a given *arbitrary* communication network (represented as a weighted, (un)directed graph). Especially, at each time-instant, a node is allowed to interact only with a neighboring node in the graph. A *randomized* gossip algorithm chooses neighbors randomly proportional to the weight of the edge connecting both. The question of interest is then how fast (if at all) such algorithm provides the global estimate to each vertex in the graph (or 'obtains a global consensus'). This task can be studied using ideas from theoretical computer science as surveyed in McDiarmid (1989), spectral graph theory (see e.g. Chung (1997)) and random walks (see e.g. Bollobas (1998)). The study of such algorithms has a long-standing history surveyed e.g. in Bertsekas and Tsitsiklis (1989), and was studied recently in e.g. Boyd et al. (2006). While a wide spectrum of computation tasks (as optimization or control) were studied in this decentralized setting, most efforts focussed essentially on first order averaging methods.

We study a randomized gossip algorithms which computes a higher-order statistic based on information available at each node. This paper restricts attention to second order U-statistics defined as follows. Consider a random variable $Z_i \in \mathcal{Z}$ following the distribution law P, and let any random variable Z_i be associated to each vertex v_i. Let $h : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ be a bounded function $\sup_{z,z' \in \mathcal{Z}} h(z, z') \leq B$, such that $h(z, z') = 0$ and $h(z, z') = h(z', z)$ for all $z, z' \in \mathcal{Z}$. Then U is defined as

$$U = \mathbb{E}_P[h(Z, Z')], \tag{1}$$

where the expectation $\mathbb{E}_P[\cdot|$ concerns the two independent copies $Z, Z' \in \mathcal{Z}$ sampled from the distribution P. Given a finite sample $\{Z_i\}_{i=1}^n$, the sample estimate U_n of U is obtained as

$$U_n = \binom{n}{2}^{-1} \sum_{i<j} h(Z_i, Z_j). \tag{2}$$

Examples include (see e.g. Serfling (1980), chapter 5):

- *Variance* [1]: Consider the case of $Z \in \mathbb{R}$ and
$$h(Z_i, Z_j) = \frac{1}{2}(Z_i - Z_j)^2. \tag{3}$$
- *Kendall's* τ. Consider the case where $Z = (X, Y)$ is a random vector with $X, Y \in \mathbb{R}$ and
$$h(Z_i, Z_j) = \text{sign}\left((X_i - X_j)(Y_i - Y_j)\right) \tag{4}$$
where $\text{sign}(z) = 1$ if $z > 0$, $\text{sign}(z) = -1$ if $z < 0$ and zero otherwise.
- *Entropy.* Consider again the case of $Z \in \mathbb{R}$
$$h(Z_i, Z_j) = \ln(|Z_i - Z_j|). \tag{5}$$
See Faivishevsky and Goldberger (2009) for properties of this estimator.

U-statistics are especially powerful when the quantities of interest can only be measured by comparing two different samples. One can for example think of the case where Z are random vectors representing complex objects as e.g. realizations of a timeseries associated to n different object. Finally, let us define the projection of U_n on a vertex v_i as

$$U_n(v_i) = \frac{1}{n} \sum_{j=1}^{n} h(Z_i, Z_j). \tag{6}$$

and $U(v_i) = \mathbb{E}_P[U_n(v_i)|Z_i]$. Note that $U_n = \frac{1}{n-1} \sum_{i=1}^n U_n(v_i)$ and consequently $U = \frac{1}{n-1} \sum_{i=1}^n U(v_i)$. Note that this quantity is directly related to the Hajék projections of U-statistics, see again Serfling (1980).

We will study a random algorithm for doing so, where the randomness comes in as each vertex will perform a random operation. If we will talk about expectation $\mathbb{E}_W$

[1] Note that this variance estimator can also be written as a double sum $\frac{1}{n} \sum_{i=1} \left(Z_i - \frac{1}{n} \sum_{j=1}^n Z_j \right)^2$, and can be implemented as such using a two stage averaging gossip algorithm.

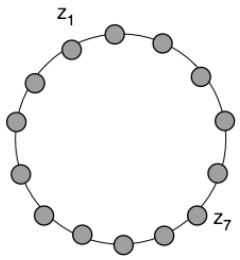

Fig. 1. Example of a communication network consisting of $n = 14$ nodes. Suppose that each node v_i has access to one random variable Z_i, the question now is how to compute $f(Z_1, \dots, Z_n)$ and circulate this value to each node v_i, using only local interactions.

or probability in the following, we will consider this conditional on the sample $\{Z_i\}_i$. In other words, expectations are taken over all possible realizations of the algorithm, not on every possible realization of the samples. The relation with resampling schemes as bootstrap however (see e.g. van der Vaart and Wellner (1996), chapter 16) suggests that additional work can tie both together.

The following notation is used throughout the paper. Let random quantities be denoted as capitals - e.g. Z - and let deterministic elements (constants, functions, variables or vectors) be denoted as lower case letters. Let 1_n denote the vector $1_n = (1, \dots, 1)^T$ of size n, and let $e_i = (0 \dots, 1, \dots, 0)^T$ be the ith unit vector. Further, we will use the notation $I_n \in \{0,1\}^{n \times n}$ for the identity matrix $I_n = \text{diag}(1, \dots, 1)$. Let $|\cdot|$ denote the cardinality of a set, or dimensionality of a vector (when appropriate). We employ vector column notation, such that for all $H \in \mathbb{R}^{n \times n}$ one has that $He_i \in \mathbb{R}^n$ yields a column with n entries. A given communication network is represented as follows. Let $G = (V, E)$ be a loopless, positively weighted graph with n vertices $V = \{v_1, \dots, v_n\}$ and edges $\{e_{ij}\}$ with corresponding positive weights $\{a_{ij}\}$. Let the *adjacency matrix* $A \in \mathbb{R}^{n \times n}$ collect those such that $A_{ij} = a_{ij}$ for all $i, j = 1, \dots, n$. Let the degree d_i of a node v_i be defined as $\sum_{j=1}^n a_{ij}$, and let the degree matrix $D \in \mathbb{R}^{n \times n}$ be defined as $D = \text{diag}(d_1, \dots, d_n)$. The Laplacian $L \in \mathbb{R}^{n \times n}$ of G is defined as

$$L = D - A, \tag{7}$$

and let $\{(\rho_i, \psi_i)\}_{i=1}^n$ denote the eigenvalues and corresponding eigenvectors $\psi_i \in \mathbb{R}^n$ of L. Note that by construction $\rho_1 = 0$ and $\psi_1 = \frac{1}{\sqrt{1_n^T D 1}} D 1_n$ (Chung (1997)).

This paper is organized as follows. Section II discusses the formal time model the gossip algorithm adopts, and a mathematical simplification of the result is motivated. The two algorithms (U1-gossip and U2-gossip) for computing $U_n(v_i)$ and U_n respectively) are studied in section III. Section IV summarizes the contribution.

2. ASYNCHRONOUS TIME MODEL

A prototypical gossip algorithm goes as follows:

(1) A randomly chosen single vertex v_i wakes up at time instant T_k and calls a randomly chosen neighbor $v_{n(i)}$;
(2) Information is exchanged between node v_i and $v_{n(i)}$;
(3) This iteration is performed sufficiently long such that consensus is guaranteed approximatively.

An appropriate (decentralized) model under which one can perform the iterations is by adopting an asynchronous time model. Here each vertex wakes up at instances $T^i = (T_1^i, T_2^i, \dots)$ described as a rate 1 Poisson process, such that the time intervals $D_k^i = T_k^i - T_{k-1}^i$ follow an exponentially distribution. This model was described e.g. in Boyd et al. (2006). Consider the expectation $\mathbb{E}_W[\cdot]$ under the stochastic algorithm as appropriate. Observe that one has that

$$\mathbb{E}_W[T_k^i] = \mathbb{E}_W\left[\sum_{l=1}^k (T_l^i - T_{l-1}^i)\right] = k, \tag{8}$$

where $T_0^i = 0$, and since the expectation of an exponential distributed random variable with rate 1 equals one. Let T denote the sorted event times of the combined n vertices, then T_k denotes the kth time something happens.

$$\mathbb{E}_W[T_k] = \mathbb{E}_W\left[\sum_{l=1}^k (T_l - T_{l-1})\right] = \frac{k}{n}, \tag{9}$$

where $T_0 = 0$, and using the fact that n independent exponential distributions yield a rate $1/n$ exponential distribution. Application of Chernoff's bound gives that one can replace T_k by $\frac{k}{n}$ if k becomes sufficiently high. On the other hand one has that a fixed node v_i is expected to call once, and to be called for once, in each time interval. Let $n_i^t \geq 0$ denote the number of times node v_i experiences an (passive or active) event up to and including iteration t. This reasoning motivates the use of two time scales, namely the absolute time denoted by τ, and t the iteration of the algorithm. As such one has

$$\mathbb{E}_W[n_i^t] = \frac{2t}{n} = 2\tau, \tag{10}$$

In the rest of the paper we will express results in terms of iterations. This is mainly for mathematical convenience. The practical implication of this convention however is that each node is supposed to know the global iteration of the algorithm. This is a global quantity, and hence somewhat in contradiction to the decentralized setup. For the sake of clarity of exposition, we content ourselves for now with the fact that each node can estimate the global iteration of the algorithm based on relation (9).

3. GOSSIP ALGORITHM FOR U-STATISTICS

3.1 Computing $U_n(v_i)$

The key design idea is to submit the random values $\{Z_i\}_i$ on a random path over the graph G (see e.g. figure 1 for an example). We will denote those objects as the *traveling* random variables. Let each node v_i have a register which contains at each time instant t exactly two random variables $Z_i \in \mathcal{Z}$ and $Z \in \{Z_1, \dots, Z_n\}$, and let the former be denoted as $Z_{(v_i)}^t \in V$. Additionally, each vertex v_i has a register $z_i^t \in \mathbb{R}$, which contains the estimate of $U_n(v_i)$ at each instance t. In summary, the state of the algorithm at iteration t and vertex v_i is

$$S_i = (Z_i, Z_{(v_i)}^t, z_i^t). \tag{11}$$

We will refer to the different entries as *the fixed register*, the *traveling register* and the *current estimate*. Let the vertices be equipped with a protocol so that they can exchange their traveling registers, denoted as

$$v_i \overset{t}{\leftrightarrow} v_k \Leftrightarrow \begin{cases} Z^t_{(v_i)} = Z^{t-1}_{(v_k)} \\ Z^t_{(v_k)} = Z^{t-1}_{(v_i)}. \end{cases} \quad (12)$$

Algorithm U2-gossip is given in Alg. 1. At each t, the estimates of $U_n(v_i)$ for all $v_i \in V$ are given as

$$\hat{U}_n(v_i) = \frac{z^t_i}{t}, \forall i = 1, \dots, n. \quad (13)$$

The remainder of this subsection proofs that $\lim_{t\to\infty} \hat{U}_n(v_i) = U_n(v_i)$ (whp).

Algorithm 1 *U*1**-gossip**

Require: Initiate the state of each vertex v_i as $S_i = (Z_i, Z_i, 0)$.

while no Consensus yet, **do**

(1) At each iteration t, a uniformly random sampled (single) node v_i wakes up and calls one of its neighbors $v_{i(k)}$, where the latter vertex is chosen randomly among the neighbors proportional to the weights of the edges, or

$$P\left(v_{n(i)} = v_k\right) = \frac{a_{ik}}{\sum_{j=1}^n a_{ij}}. \quad (14)$$

(2) Then $v_i \overset{t}{\leftrightarrow} v_{n(i)}$

(3) Then the current estimates are updated for any $i = 1, \dots, n$ as

$$z^t_i = z^{t-1}_i + h(Z_i, Z^t_{(v_i)}). \quad (15)$$

end while

This algorithm is defined in terms of a sequence of exchange operations. I.e., at event time T_k, a node v_i wakes up and calls v_j. They then exchange a piece of information. This can be written down in terms of a permutation matrix $W_{(ij)}$ where $W_{(ij),kk} = 1$ for all $k \neq i, j$, $W_{(ij),ii} = W_{(ij),jj} = 0$ and $W_{(ij),ij} = W_{(ij),ji} = 1$ and zero otherwise, or schematically

$$W_{(ij)} = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 0 & \dots & 1 & \\ & & \vdots & \ddots & \vdots & \\ & & 1 & \dots & 0 & \\ & & & & & \ddots \\ & & & & & & 1 \end{bmatrix} \quad (16)$$

If e^t denotes the unit vector with $e^t_i = 1$ if at instance t the object is located at v_i, one can write after step $v_i \overset{t}{\leftrightarrow} v_j$ that $e^t = W_{(ij)} e^{t-1}$. Let $W^{(1)}, W^{(1)}, \dots, W^{(1)}$ be t independent random permutation matrices, then

$$e^t = \left(W^{(t)} \dots W^{(2)} W^{(1)}\right) e^0. \quad (17)$$

following the protocol. The expected matrix denoted as $W \in [0,1]^{n\times n}$ becomes

$$W = \mathbb{E}_W[W_{(ij)}] = \frac{1}{n} \sum_{i=1}^n \left(\sum_{j=1}^n \frac{a_{ij}}{\sum_{j=1}^n a_{ij}} W_{(ij)} \right). \quad (18)$$

As W is a linear combination of permutation matrices, W is a doubly stochastic matrix $W \geq 0$, $1^T_n W = 1^T_n$ and $W 1^T_n = 1^T_n$. A realization of an algorithm is hence fully characterized by a sequence of realizations of W. As in theorem 3 of Boyd et al. (2006), one has that

Proposition 1. (The Mixing matrix W).

$$W = \frac{1}{n} \sum_{ij} \frac{a_{ij}}{d_i} W_{(ij)} = I_n - \frac{1}{n}(D' - A'). \quad (19)$$

where $A' = \left(D^{-1}A + (D^{-1}A)^T\right)$ and where $D' = \text{diag}\left(A' 1^T_n\right)$.

Note that W is symmetric positive definite regardless of the design of A. As a doubly stochastic matrix, it has eigenvalues $1 = \lambda_1 > \lambda_2 \geq \dots \lambda_n \geq 0$. Let $\phi_i \in \mathbb{R}^n$ denote the eigenvectors associated to those values.

At this point, we investigate a bit deeper the relation between the eigenvalues of W and of the communication graph G. We restrict attention to the case of undirected, positively weighted, regular graphs. This means that the degrees of all edges are identical, or $d_1 = \dots = d_n = d$. Regularity of the graph is no restriction as for any positively weighted graph with adjacency matrix A, one can make a graph with adjacency matrix $\frac{1}{2}(D^{-1}A - A^T D^{-T})$ which has constant degree 1. In this case, eq. (19) reduces to $W = I_n - \frac{2}{dn} L$. Let $(\rho_i, \psi_i)_i$ the eigenvalues and -vectors of the Laplacian L, then it is follows that

Proposition 2. (Eigenvalues of W).

$$\lambda_i = 1 - \frac{2}{dn}\rho_i, \quad \phi_i = \psi_i, \ \forall i = 1, \dots, n. \quad (20)$$

Using this notation, one can write that the expected position of a particle which was in e^0 would be $\mathbb{E}_W[e^t]$ given as

$$\mathbb{E}_W[e^t] = \mathbb{E}_W\left[W^{(t)} \dots W^{(2)} W^{(1)}\right] e^0 = W^t e^0. \quad (21)$$

after t iterations. This follows from the independence of the different iterations of the algorithm. Now, using the fact that one can represent $e^0 = \sum_{i=1}^n \alpha_i \phi_i$ for an appropriate vector $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{R}^n$ where $\|\alpha\|_2 = 1$, one has that

$$\mathbb{E}_W[e^t] - \frac{1}{n} \leq \sum_{i=2}^n \alpha_i \lambda_i^t \leq (n-1)\exp(-(1-\lambda_2)t), \quad (22)$$

where we use the inequality $\ln(1-x) \leq -x$ for all $x \in (0, 1[$. This result implies that the algorithm will distribute the elements evenly over all vertices with an exponential rate governed by $(1-\lambda_2)$. Then the intuition behind the algorithm is that taking expectation w.r.t $h(Z_i, \cdot)$ equals $U_n(v_i)$ as in eq. (6) as desired. In order to let the variance of the estimate also decrease with t, we let the algorithm average the result over all iterations so far, resulting in the cumulative rule (15). We now make this reasoning more formal.

Let the vector $h_i \in \mathbb{R}^n$ be defined as $h_{i,j} = h(Z_i, Z_j)$ for all $j = 1, \dots, n$. The expectation of the current estimate z^t_i can be written as follows

Proposition 3. (**Expectation of z^t_i**). After t iterations of the algorithm (or roughly t/n units of time, see subsection 2.1) one has

$$\mathbb{E}_W\left[z^t_i\right] = \sum_{l=1}^t h_i W^l e^T_i. \quad (23)$$

Proof: By additivity of the expectation, one can write that after the t-th iteration of the algorithm one has

$$\mathbb{E}_W\left[z_i^t\right] = \mathbb{E}_W\left[z_i^{t-1}\right] + \sum_{j=1}^{n} h(Z_i, Z_j) P(Z_{(v_i)}^t = Z_j)$$
$$= \mathbb{E}_W\left[z_i^{t-1}\right] + \sum_{j=1}^{n} h(Z_i, Z_j)(e_j W^t e_i)$$
$$= \sum_{l=1}^{t} h_i W^l e_i^T, \quad (24)$$

as the term $e_k W^t e_i$ represents the probability of item Z_k being in $Z_{(v_i)}^t$ after t iterations when starting with $Z_{(v_i)}^0 = Z_i$. Then by unravelling the recursion the result is proven.

□

This elementary result is important as it casts the algorithmic analysis in a framework of matrix algebra. We now proceed relating $\mathbb{E}_W\left[z_i^t\right]$ to $U_n(v_i)$. Therefore we need the following technical result which will govern the convergence of the dynamical system underlying the algorithm.

Proposition 4. (**Geometric series).** One has for $\lambda \in]-1, 1[$ and $k > 0$ that

$$\left|\frac{1}{t}\sum_{l=1}^{t} \lambda^l\right| \leq \left|\frac{(n-1)\lambda}{(1-\lambda)t}\right|. \quad (25)$$

This result is easily seen by considering the geometric series $\frac{1}{1-x} = 1 + x + x^2 + x^3 + \ldots$ for all $x \in]-1, 1[$. Observe that the right hand side can grow unboundedly when $\lambda \to 1$.

Lemma 1. (**Unbiasedness of $U1$-gossip).** Consider the estimates z_i^t of the algorithm after t iterations. Then one has

$$\mathbb{E}_W\left|\frac{z_i^t}{t} - U_n(v_i)\right| \leq \left|\frac{(n-1)\lambda_2}{(1-\lambda_2)t}\right| \|h_i\|_2. \quad (26)$$

Proof: Note again that

$$U_n(v_i) = \frac{1}{n}\sum_{i=1}^{n} h(Z_i, Z_j) = \frac{1}{\sqrt{n}}\phi_1^T h_i, \quad (27)$$

by definition. This term converges to zero by the convergence of a Markov chain to its stationary distribution, shown as follows. Then one can express any vector e_i as

$$e_j = \sum_{i=1}^{n} \alpha_i^j \phi_i, \quad (28)$$

with appropriate values $\alpha^j = (\alpha_1^j, \ldots, \alpha_n^j)^T \in \mathbb{R}^n$, and $\|a^j\|_2 = 1$. Hence $e_j = \sum_{i=1}^n \alpha_i^j \lambda_i^l \phi_i$ for all $j = 1, \ldots, n$. The value of α_1^j for all $j = 1, \ldots, n$ is given as follows

$$\alpha_1^j = \phi_1^T e_j = \frac{1}{\sqrt{n}}. \quad (29)$$

Modifying slightly the classical proof of convergence as given in Bollobas (1998), theorem 27, or Chung (1997), subs. 1.5 gives

$$\mathbb{E}_W\left|\frac{1}{t}z_i^t - U_n(v_i)\right| = \left|\frac{1}{t}\sum_{l=1}^{t} e_j^T W^l h_i - \frac{1}{\sqrt{n}}\phi_1^T h_1\right|$$
$$= \left|\frac{1}{t}\sum_{l=1}^{t}\sum_{i=1}^{n} \alpha_i^j \lambda_i^l \phi_i^T h_i - \frac{1}{\sqrt{n}}\phi_1^T h_1\right|$$
$$= \left|\alpha_1\phi_1^T h_i + \frac{1}{t}\sum_{l=1}^{t}\sum_{i=2}^{n} \alpha_i^j \lambda_i^l \phi_i^T h_i - \frac{1}{\sqrt{n}}\phi_1^T h_1\right|$$
$$= \left|\frac{1}{t}\sum_{i=2}^{n} \alpha_i^j \sum_{l=1}^{t} \lambda_i^l \phi_i^T h_i\right|$$
$$\leq \left\|\frac{1}{t}\sum_{i=2}^{n} \alpha_i^j \sum_{l=1}^{t} \lambda_i^l \phi_i\right\|_2 \|h_i\|_2$$
$$\leq \frac{n-1}{t}\sum_{l=1}^{t} \lambda_2^l \|h_i\|_2 \|\alpha^j\|_2 \leq \left|\frac{(n-1)\lambda_2}{(1-\lambda_2)t}\right| \|h_i\|_2 \quad (30)$$

using respectively the eigenvalue representations (40) and (29), Cauchy-Schwartz' inequality and in the final step (4) the inequality (25).

□

Now we bound the difference between the actual terms z_i^t and the expected value $E[z_i^t]$ by application of a bounded difference probabilistic inequality. Our concentration result will depend on McDiarmid's bounded difference inequality. We here use the version as in McDiarmid (1989), corollary 6.10, which requires slightly weaker conditions than the standard bounded difference inequality. This altered conditions will exactly match our requirements.

Lemma 2. (**Bounded Difference Inequality).** Let $X = (X_1, \ldots, X_n) \in \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ be a random vector with n elements in $X_i \in \mathcal{A}_i$, and let $f : \mathcal{A}_1 \times \cdots \times \mathcal{A}_n \to \mathbb{R}$ be an appropriately measurable function. Suppose that there are constants $c_1, \ldots, c_n$ such that for all $i = 1, \ldots, n$ one has

$$\sup_{x_1,\ldots,x_i,x_i'} \Big|E[f(X)|(X_1, \ldots, X_i) = (x_1, \ldots, , x_{i-1}, x_i)]$$
$$- E[f(X)|(X_1, \ldots, X_i) = (x_1, \ldots, x_{i-1}, x_i')]\Big| \leq c_i, \quad (31)$$

where for all $i = 1, \ldots, n$ one has $(x_1, \ldots, x_i) \in \mathcal{A}_1 \times \cdots \times \mathcal{A}_i$ and $x_i' \in \mathcal{A}_i$. Then one has for all $\epsilon > 0$ that

$$P\left(|E[f(X)] - f(X)| \geq \epsilon\right) \leq \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^n c_i^2}\right). \quad (32)$$

We will apply this on the function $f(\sigma_i) = z_i^l$ giving the following result.

Lemma 3. (**Concentration of $U1$-gossip).** Fix $\delta > 0$ and $k > 0$, and consider a time instant t such that $n_i^t \geq k$. Then one has with probability exceeding $1 - \delta$ that

$$\left|\mathbb{E}_W\left[\frac{z_i^t}{t}\right] - \frac{z_i^t}{t}\right| \leq \left(\sqrt{2}n\left(\frac{1}{1-\lambda_2}\right)\|h_i\|_2\right)\sqrt{\frac{\ln(2/\delta)}{t}}. \quad (33)$$

Proof: We express the conditional probability of z_i^t given the history until just before iteration k. Let $e_{(v_i)}^k$ be the unit vector with nonzero element at the index of the random variable which is in the variable register of v_i at iteration k. Then we can write the expectation of z_i^t conditioned on the history before iteration k as

$$\mathbb{E}_W\left[z_i^t\middle|z_i^{k-1},W^{(k)},e_{(v_i)}^{k-1}\right]=z_i^{k-1}+e_{(v_i)}^{k-1}W^{(k)}h_i+\frac{1}{n}\sum_{j=1}^{t-k}e_{(v_i)}^{k-1}W^{(k)}\mathbb{E}_W\left[W^{(k+1)}\dots W^{(k+j)}\right]h_i=z_i^{k-1}+\frac{1}{n}\sum_{j=0}^{t-k}e_{(v_i)}^{k-1}W^{(k)}W^jh_i \tag{34}$$

Specify the function f in the previous Lemma as z_i^t, then we have that

$$\mathbb{E}_W\left[z_i^t\middle|W^{(k)},z_i^{k-1},e_{(v_i)}^{k-1}\right]-\mathbb{E}_W\left[z_i^t\middle|W^{(k)\prime},z_i^{k-1},e_{(v_i)}^{k-1}\right]=\sum_{j=0}^{k-l}v_k^TW^jh_i. \tag{35}$$

where we define the vector $v_k\in\{0,1\}^n$ as

$$v_k=\left(e_{(v_i)}^kW^{(k)}-e_{(v_i)}^kW^{(k)\prime}\right). \tag{36}$$

Now note that we can assume that $i_l\neq i_l'$ since otherwise the difference would be zero. This implies that $1_n^T\left(e_{i_l}-e_{i_l'}\right)=0$, and the vector $\left(e_{i_l}-e_{i_l'}\right)$ will be not in the span of λ_1. As such we have

$$\begin{aligned}\text{eq. (36)}&\leq\sqrt{2}\sup_{v:\|v\|_2\leq1,v^T1_n=0}\sum_{j=1}^{k-l}v^TW^{(k)}W^jh_i\\&\leq\sqrt{2}\sup_{v:\|v\|_2\leq1}\sum_{j=1}^{k-l}\sum_{s=2}^{n}\lambda_s^jv^T(\phi_s\phi_s^T)h_i\\&\leq\sqrt{2}\sup_{v:\|v\|_2\leq1}\sum_{j=1}^{k-l}\sum_{s=2}^{n}\lambda_s^jv^Th_i\\&\leq\sqrt{2}n\left(\frac{1}{1-\lambda_2}\right)\|h_i\|_2\end{aligned} \tag{37}$$

by application of Proposition 2, boundedness of the kernel h. This proves the bounded difference inequalities, and as such Lemma 2 can be applied, yielding for any $\epsilon>0$ that

$$P\left(\left|\mathbb{E}_W\left[\frac{z_i^t}{t}\right]-\frac{z_i^t}{t}\right|\geq\epsilon\right)\leq\exp\left(-\frac{2\epsilon^2t}{\left(\sqrt{2}n\left(\frac{1}{1-\lambda_2}\right)\|h_i\|_2\right)^2}\right). \tag{38}$$

□

Now we are ready to prove our first main result.

Theorem 1. (**Convergence of $U1$-gossip).** Let $\delta>0$. When executing the algorithm, the current estimate $\frac{z_i^t}{n_i^t}$ will approximate $U_n(v_i)$ such that with probability exceeding $1-\delta$ one has

$$\sup_{v_i\in V}\left|\frac{z_i^t}{t}-U_n(v_i)\right|\leq\left(\sqrt{2}n\left(\frac{1}{1-\lambda_2}\right)\|h_i\|_2\right)\sqrt{\frac{\ln(2n/\delta)}{t}}+\left|\frac{(n-1)\lambda_2}{(1-\lambda_2)t}\right|\|h_i\|_2. \tag{39}$$

Proof: Consider first the situation for a fixed vertex $v_i\in V$, and consider the following decomposition

$$\left|\frac{z_i^t}{t}-U_n(v_i)\right|\leq\left|\frac{z_i^t}{t}-\mathbb{E}_W\left[\frac{z_i^t}{t}\right]\right|+\left|\mathbb{E}_W\left[\frac{z_i^t}{t}\right]-U_n(v_i)\right|. \tag{40}$$

The last term $|\mathbb{E}_W[z_i^t/t]-U_n(v_i)|$ in the decomposition (40) is proven in Lemma 3. The first term $|z_i^t/t-\mathbb{E}_W[z_i^t/t]|$ in (40) is proven by the concentration result of Lemma 2. Appending the two inequalities (33) and (30), adding a Bonferroni correction for the result to hold over all vertices v_i simultaneously gives the desired result.

□

Observe that this bound becomes arbitrary loose when λ_2 goes to one. Following proposition 2, this means that ρ_2 would be close to zero, which in turn indicates a loosely connected graph. In this case, it is seen that convergence to the stationary distribution can take arbitrary long. If λ_2 is sufficiently high - or the graph sufficiently well-connected - the approximation properties of this algorithm decays roughly as $O(1/\sqrt{t})$.

3.2 Computing U_n

We now turn attention to compute U_n and to have it available at all nodes V simultaneously, again without performing global computations. A simple approach here would consist in computing the quantities $\{U_n(v_i)\}_i$ in a first step using $U1$-gossip, and subsequently average this quantities out using a standard gossip algorithm. This approach is however suboptimal as the two stages can be merged together in one iteration. In order to implement this idea, let us first extend the states of a vertex v_i at instance t. We let the fixed register in S_i of eq. (11) be replaced by a second traveling register $Z_{[v_i]}^t$, or

$$S_i'=\left(Z_{(v_i)}^t,Z_{[v_i]}^t,z_i^t\right). \tag{41}$$

Now the algorithm sends both traveling random variables on an independent random walk, such that $\mathbb{E}_W[h(Z_{(v_i)},Z_{[v_i]})]\approx U_n$ for any vertex $v_i\in V$. The algorithm is given in Alg. 2. The final estimate of U_n at vertex v_i is

$$\hat{U}_n^i=\left(\frac{n}{n-1}\right)\frac{z_i^t}{t}. \tag{42}$$

The analysis of this algorithm goes similar as the one of U1-gossip. Again, the crucial relation is

$$\mathbb{E}_W\left[\frac{z_i^t}{t}\right]=\frac{1}{t}\sum_{l=1}^{t}e_i^TW^lHW^le_i, \tag{45}$$

using the notation as before. Then this approximates for any $i=1,\dots,n$

$$\frac{n}{n-1}U_n=\frac{1}{n^2}1_n^TH1_n, \tag{46}$$

as $W^te_i\to\frac{1}{n}1_n$. Formally,

Lemma 4. (**Unbiasedness of $U2$-gossip).** After t iterations of the algorithm U2-gossip, one has that

$$\left|U_n-\mathbb{E}_W[\hat{U}_n^i]\right|\leq\left|\frac{n\lambda_2^2}{(1-\lambda_2^2)t}\right|\|H\|_2. \tag{47}$$

Proof: Let again $e_i=\sum_{j=1}^n\alpha_j^i\phi_j$ with $\|\alpha^i\|_2=1$. Since $\phi_1=\frac{1}{\sqrt{n}}1_n$ and $\lambda_1=1$, one has that

Algorithm 2 $U2$**-gossip**

Require: Initiate the state of each vertex v_i as $S_i' = (Z_i, Z_i, 0)$

while no Consensus yet, **do**

(1) At each iteration t, a uniformly random sampled (single) node v_i wakes up and calls one of its neighbors $v_{n(i)}$, where the latter vertex is chosen randomly among the neighbors proportional to the weights of the edges, or

$$P\left(v_{n(i)} = v_k\right) = \frac{a_{ik}}{\sum_{j=1}^{n} a_{ij}}. \tag{43}$$

(2) Then $v_i \overset{t}{\leftrightarrow} v_{n(i)}$

(3) At the same iteration, but independently a node v_j is selected, it contacts a neighbor $v_{n(j)}$ using (43), one has $Z^t_{[v_j]} = Z^{t-1}_{[v_{n(j)}]}$ and $Z^t_{[v_{n(j)}]} = Z^{t-1}_{[v_j]}$.

(4) Then the current estimates are updated for any $i = 1, \ldots, n$ as

$$z_i^t = z_i^{t-1} + h(Z^t_{(v_i)}, Z^t_{[v_i]}). \tag{44}$$

end while

$$\left|\frac{1}{n}\phi_i^T H \phi_i - \frac{1}{t}\sum_{l=1}^{t} e_i^T W^l H W^l e_i\right| \leq \frac{1}{t}\left|\sum_{l=1}^{t}\sum_{j=2}^{n} \alpha_j^i \lambda_i^{2l}\right| \|H\|_2 \leq \left|\frac{(n-1)\lambda_2^2}{(1-\lambda_2^2)t}\right| \|H\|_2, \tag{48}$$

using Cauchy-Schwarz and Proposition 3.

□

remark that now the mixing is governed by λ_2^2, which is in general lower (and hence slower) than λ_2. It can also be seen that the estimate $\hat{U}_n$ concentrates around its mean U_n. This result makes again use of Lemma 2.

Lemma 5. (**Concentration of** $U2$**-gossip).**

$$P\left(\left|\mathbb{E}_W\left[\hat{U}_n^i\right] - \hat{U}_n^i\right| \geq \epsilon\right) \leq \exp\left(-\frac{2\epsilon^2 t}{\left(\sqrt{2}n\left(\frac{1}{1-\lambda_2^2}\right)\|H\|_2\right)^2}\right). \tag{49}$$

The proof is entirely analogue to the proof of Lemma 3. This implies the following convergence proof for the U2-gossip algorithm.

Theorem 2. (**Convergence of** $U2$**-gossip).** Let $\delta > 0$, and iterate the algorithm at least t times. Then with probability exceeding $1 - \delta$ one has

$$\sup_{v_i \in V}\left|\hat{U}_n^i - U_n\right| \leq \left(\sqrt{2}n\left(\frac{1}{1-\lambda_2^2}\right)\|H\|_2\right)\sqrt{\frac{\ln(2n/\delta)}{t}} + \left|\frac{n\lambda_2^2}{(1-\lambda_2^2)t}\right|\|H\|_2. \tag{50}$$

Note again that in the case the graph G is regular with ρ_2 the second smallest eigenvalue, one sees that convergence takes place with $O(1/(\sqrt{t}\rho_2^2))$. This bound grows when the communication network consists of different almost disconnected parts.

4. CONCLUSION

This manuscript [2] analyzes two decentralized, randomized 'gossip' algorithms for computing U-statistics. U1-gossip computes the projections (6) on each vertex, and U2-gossip computes the global U_n as in (2). The analysis of both is based on the convergence of an associated Markov chain, governed by the Fiedler value of the communication graph, and the use of a martingale concentration inequality.

There remain a couple of issues to be coped with formally. The first one is that the algorithm needs to have knowledge of the times and number of all past iterations. As suggested in Subsection 2.1., this could be handled by replacing the denominator t in (39) and (50) by a random variable, denoting an estimate of the number of iterations instead. This prompts in turn a more involved analysis concerning convergence of the ratio of random variables. Another open question is how to find matching lowerbounds for the algorithm, requiring further results on martingale inequalities.

REFERENCES

Bertsekas, D. and Tsitsiklis, J. (1989). *Parallel and distributed computation.* Prentice Hall.

Bollobas, B. (1998). *Modern graph theory.* Springer.

Boyd, S., Ghosh, A., Prabhakar, B., and Shah, D. (2006). Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6), 2508–2530.

Chung, F. (1997). Spectral Graph Theory.

Faivishevsky, L. and Goldberger, J. (2009). ICA based on a smooth estimation of the differential entropy. *Proceedings of the 22 Annual Conference on Neural Information Processing Systems (NIPS22).*

McDiarmid, C. (1989). On the method of bounded differences. *Surveys in combinatorics*, 141, 148–188.

Serfling, R. (1980). *Approximation Theorems of Mathematical Statistics.* John Wiley & Sons.

van der Vaart, A. and Wellner, J. (1996). *Weak Convergence and Empirical Processes.* Springer.

[2] ACKNOWLEDGMENTS - KP is a assistant professor ("forskarsassistent") in the Department of Information Technology in Uppsala University, Sweden. JS is a professor at the Katholieke Universiteit Leuven, Belgium. Research supported by (Research Council KUL) GOA AMBioRICS, CoE EF/05/006 Optimization in Engineering(OPTEC), IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; (Flemish Government FWO) PhD/postdoc grants, projects G.0452.04 (new quantum algorithms), G.0499.04 (Statistics), G.0211.05 (Nonlinear), G.0226.06 (cooperative systems and optimization), G.0321.06 (Tensors), G.0302.07 (SVM/Kernel), G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine) research communities (ICCoS, ANMMM, MLDM); IWT: PhD Grants, McKnow-E, Eureka-Flite+ (Belgian Federal Science Policy Office) IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011) ; (EU) ERNSI; FP7-HD-MPC (Collaborative Project STREP-grantnr. 223854) Contract Research: AMINAL; Other: Helmholtz: viCERP.