King's Research Portal

Document Version
Peer reviewed version

Download date: 15. Jan. 2025

Distributed Averaging in Opinion Dynamics

P. Berenbrink
Universität Hamburg
Hamburg, Germany

C. Cooper
King's College London
London, United Kingdom

C. Gava
King's College London
London, United Kingdom

F. Mallmann-Trenn
King's College London
London, United Kingdom

T. Radzik
King's College London
London, United Kingdom

D. Kohan Marzagão
University of Oxford
Oxford, United Kingdom

N. Rivera
University of Valparaíso
Valparaíso, Chile

ABSTRACT

We consider two simple asynchronous opinion dynamics on arbitrary graphs where every node u of the graph has an initial value $\xi_u(0)$. In the first process, which we call the *NodeModel*, at each time step $t \geq 0$, a random node u and a random sample of k of its neighbours $v_1, v_2, \cdots, v_k$ are selected. Then, u updates its current value $\xi_u(t)$ to $\xi_u(t+1) = \alpha \xi_u(t) + \frac{(1-\alpha)}{k} \sum_{i=1}^{k} \xi_{v_i}(t)$, where $\alpha \in (0,1)$ and $k \geq 1$ are parameters of the process. In the second process, called the *EdgeModel*, at each step a random pair of adjacent nodes (u, v) is selected, and then node u updates its value equivalently to the *NodeModel* with $k = 1$ and v as the selected neighbour.

For both processes, the values of all nodes converge to the same value F, which is a random variable depending on the random choices made in each step. For the *NodeModel* and regular graphs, and for the *EdgeModel* and arbitrary graphs, the expectation of F is the average of the initial values $\frac{1}{n} \sum_{u \in V} \xi_u(0)$. For the *NodeModel* and non-regular graphs, the expectation of F is the degree-weighted average of the initial values.

Our results are two-fold. We consider the concentration of F and show tight bounds on the variance of F for regular graphs. We show that when the initial values do not depend on the number of nodes, then the variance is negligible, and hence the nodes are able to estimate the initial average of the node values. Interestingly, this variance does not depend on the graph structure. For the proof we introduce a duality between our processes and a process of two correlated random walks. We also analyse the convergence time for both models and for arbitrary graphs, showing bounds on the time T_ε required to make all node values 'ε-close' to each other. Our bounds are asymptotically tight under some assumptions on the distribution of the initial values.

CCS CONCEPTS

• **Theory of computation → Random walks and Markov chains**; **Random network models**; **Distributed algorithms**; • **Mathematics of computing → Markov processes**.

KEYWORDS

Distributed Averaging, Dual Processes, Stochastic Processes, Random Walks

1 INTRODUCTION

In this paper, we consider two natural asynchronous opinion dynamics, extending the notion of pull voting to averaging of numeric opinions. We refer to these models as the *NodeModel* and the *EdgeModel*. We are given a connected undirected graph $G = (V, E)$ with n nodes, which we refer to as agents, and m edges. Each agent u has a real number as its initial value at time 0. In the *NodeModel*, at each time step $t \geq 1$, a node u is chosen uniformly at random. This node observes $k \geq 1$ of its neighbours $v_1, \ldots v_k$, selected uniformly at random, and updates its value unilaterally (i.e., only u updates its value at a this step) to an α-fraction of its current value plus a $(1 - \alpha)$-fraction of the average of the values of $v_1, \ldots v_k$. In the *EdgeModel*, a (directed) edge (u, v) is observed uniformly at random and u updates its value unilaterally to an α-fraction of its own value plus a $(1 - \alpha)$-fraction of the value of v. For regular graphs and $k = 1$ the *EdgeModel* is the same as the *NodeModel*. Similarly to the voter model, where a randomly selected node takes on the opinion of a random neighbour, the dynamics which we consider are very natural, possibly the simplest randomised pull-based protocols for reaching consensus on an opinion which is close to the average of the initial opinions. Agents in this protocols do not perform any complicated processing of the information obtained, other than the computation of a simple average of the received values.

In contrast to neighbourhood load balancing processes, in our model, only one selected node u changes its value at each time step.

This is another aspect of the simplicity of our processes, as they do not require coordination of simultaneous updates in two or more nodes.

It is easy to see that, over time, the nodes' values converge to a single value, which we denote by F and refer to as the *convergence value*. This follows from observing that $\max_{u,v \in V} |\xi_u(t) - \xi_v(t)|$, where $\xi_x(t)$ is the value at node x at the end of step t, is non-increasing and tends to 0. F is a random variable (depending on the random choices of the protocols) and its expectation turns out to be equal to the average of the initial values $\frac{1}{n} \sum_{u \in V} \xi_u(0)$ in the *EdgeModel*, and the degree-weighted average of the initial values $\frac{1}{2m} \sum_{u \in V} d_u \xi_u(0)$ in the *NodeModel*. Hence we refer to either of these models as an Averaging Process.

A natural question is to quantify the variance of F, $\mathbb{V}ar(F)$, to understand when F is concentrated around its expected value. Estimating the variance and analysing the convergence rate of our protocols are the main aims of this paper. By quantifying the variance of F, we analyse the ability of our processes to estimate the initial average of the node values. In comparison to load balancing processes, which keep the average node load as invariant, divergence of F from its expectation can be viewed as the price of simplicity of our averaging processes.

To bound $\mathbb{V}ar(F)$ we introduce a time-reversed dual process. This process simplifies the analysis and enables us to derive the exact value of the variance for regular graphs, which, under some mild assumptions on the initial distribution of the node values, implies concentration of F. Regarding the convergence time, we derive tight bounds for both protocols by using an appropriate potential function. These results are summarised in Theorem 2.2, with a more detailed statement regarding $\mathbb{V}ar(F)$ given in Proposition 5.8. In the *NodeModel* our bounds on both the convergence time and the $\mathbb{V}ar(F)$ show, somewhat surprisingly, only a negligible dependence on k. It makes almost no difference if $k = 1$ or if it is close to the node degree. Another interesting property is that $\mathbb{V}ar(F)$ is independent (up to some constants) of both the graph structure and the node mapping of the initial values. For example, for the same set of initial node values and *NodeModel* with $k = 1$, the variance $\mathbb{V}ar(F)$ is the same for complete graphs as for cycle graphs.

Processes with unilateral updates can provide a natural model in various scenarios. The graph can represent a social network where individuals (the nodes) change their opinions after observing opinions of some of their friends. For example, they might want to decide which phone to buy or how much they should budget for a given type of vacation, and check their friends' opinion on this. Unilateral updates model situations where an individual is influenced by the opinion of specialist friends. This does not necessarily mean that a "specialist" is also influenced by the opinions of all its friends (e.g. see [8]). There are numerous examples for models with unilateral updates in different areas, including: population dynamics [39], biology [16] and robotics [7].

Note that the simple pull-based communication with unilateral updates considered in this paper cannot guarantee convergence to the initial (simple or weighted) average. To guarantee such convergence, one would need a stronger communication model. For example, it is well-known that if two nodes average their values

in a given time step (e.g., two neighbours in the graph in a load-balancing process), then convergence to the average is guaranteed [2]. Such a protocol, however, requires coordinated updates at these two nodes. Compare this with our models where we do not have coordinated updates. In this context, analysing the variance of F can be viewed as studying the cost of simplicity.

Our Results. In this paper we show tight bounds on the convergence time of the *NodeModel* and the *EdgeModel*. Furthermore, we calculate the variance of the final value F at the nodes. The update at step $t \geq 1$ in both models can be expressed as $\xi(t) = B(t)\xi(t-1)$, where $B(t) \in \mathbb{R}^{|V| \times |V|}$ is the *communication matrix* for step t and $\xi(t) \in \mathbb{R}^{|V|}$ is the vector of node values at the end of this step. There are many convergence results for models which have *doubly-stochastic* (average-preserving) update matrices (e.g., [22]). In contrast, the update matrices in our unilateral-update models do not have this property (matrices $B(t)$ are stochastic, but not doubly-stochastic) posing challenges in analysis. There are far less results for this setting [31]. Note, for example, that in our models the convergence value is not known ahead of time, requiring a novel potential function, especially for non-regular graphs.[1]

The main technical novelty of our paper lies, however, in the development of methods for obtaining asymptotically tight concentration bounds (as detailed in Proposition 5.8). A well-known duality between the voter model and coalescing random walks,[2] obtained by coupling these two processes (one being run backwards in time), shows that the voting time and the coalescence time have the same distribution. We derive a generalisation of this duality which relates the *NodeModel* to a diffusion process. In this dual diffusion process, we have n different 'commodities', indexed by the nodes, and start by placing the unit amount of commodity i on node i, for each $i \in V$. These commodities are then being diffused throughout the graph in the following steps. In one step, first a node u and its k neighbours are randomly selected. Then for each commodity which currently has positive 'load' at node u, a $(1-\alpha)$ fraction of this load is taken from u and distributed in equal proportion among the selected k neighbours.

If we couple T steps of the *NodeModel* process with the above diffusion process, running one of them backwards in time, then at the end of these T steps, the distribution of node values in *NodeModel* is the same as the (weighted) distribution of commodities in the diffusion process. Further, we show that the variance of the diffusion process is the same as the variance of a process of two correlated random walks associated with the diffusion process. The latter random walk process can be analysed by studying the stationary distribution of an associated Markov chain. Crucially, despite the fact that this Markov chain is not bi-directional (and so, it does not satisfy the detailed balanced equations), we are able to compute its exact stationary distribution. Using this distribution and arguing via the diffusion process, we obtain the variance of the convergence value of the *NodeModel*. The formal definitions, the formal statement of the duality and the details of the analysis are given in Section 5.

---

[1] The standard potential functions which do not take the degrees into account are not decreasing in expectation.

[2] In coalescing random walks, one random walk starts on each node and whenever two or more random walks meet, they merge into one random walk. The coalescence time is the expected time until only one random walk is left.

Some of the proofs are omitted, but they are included in the full version of this paper [9].

2 THE MODEL AND DETAILED CONTRIBUTION

We have a connected undirected graph $G = (V, E)$ with n nodes, m edges and the set of nodes $V = \{1, 2, \ldots, n\}$. If the graph is regular we use d for the node degree; otherwise d_i is the degree of node i and $d_{\max}$ and $d_{\min}$ are the maximum and minimum node degree. We assume that every node has an initial *value*, which is a real number. Furthermore, $\xi_i(t)$ is defined as a random variable equal to the value of node $i \in V$ at the end of the t-th time step, and $\xi(t)$ as the vector of these values. The vector $\xi(0) = \xi$ represents the nodes' initial values. In this paper we show results for two different models called *NodeModel* and *EdgeModel*. In the following we will first introduce the *NodeModel* and state the corresponding results. Then, we will do the same for the *EdgeModel*.

DEFINITION 2.1 (*NodeModel*). *The model has two parameters: a real number $\alpha \in [0, 1)$ and an integer $k \geq 1$. At each time step $t \geq 1$, a node u is chosen uniformly at random. That node, in turn, chooses k of its neighbours $v_1, v_2, \ldots v_k$ uniformly at random and without replacement. Then,*

$$\xi_u(t) = \alpha \cdot \xi_u(t-1) + \frac{(1-\alpha)}{k} \cdot \sum_{i=1}^{k} \xi_{v_i}(t-1),$$

and for all $v \neq u$, $\xi_u(t+1) = \xi_u(t)$.

Note that for $k = 1$ and $\alpha = 0$ this model is equivalent to the voter model, where a node chosen uniformly at random takes the opinion of its random neighbour. It is easy to see that in the *NodeModel* the values of nodes converge to the same value F. In Lemma 4.1, we show that $\mathbb{E}(F) = \sum_{i=1}^{n} \frac{d_i}{2m} \cdot \xi_i(0)$. In the following we assume w.l.o.g. that the initial values are *centered at 0*, so that $\sum_{i=1}^{n} \frac{d_i}{2m} \cdot \xi_i(0) = 0$. For regular graphs, this becomes $\mathbb{E}(F) = \frac{1}{n} \sum_i \xi_i(0)) = 0$. Let $1 - \lambda_2(P)$ be the eigenvalue gap of the associated transition matrix P (defined in Section 4), and for $x \in R^n$, let $\|x\|_2$ be the Euclidean norm of x.

THEOREM 2.2 (*NodeModel*). *Let G be a connected graph, α a constant in $(0, 1)$ and $k \geq 1$ integer (parameters of the model). Let $\varepsilon \in (0, 1)$, and let $T = T_\varepsilon$ be the first time such that $\xi(T)$ is ε-converged (defined in Section 4).*

(1) (Convergence time) Then, we have w.h.p.

$$T = O\left(\frac{n \log\left(n\|\xi(0)\|_2^2/\varepsilon\right)}{1 - \lambda_2(P)}\right).$$

Furthermore, there exists an initial state $\xi(0)$ for which this bound is, up to constants, tight for every $k \geq 1$.

(2) (Concentration) If G is regular and regardless of k, we have

$$\mathbb{V}ar(F) = \Theta\left(\frac{\|\xi(0)\|_2^2}{n^2}\right).$$

The proof of the first part of the theorem can be found in the full version, where we also present some more detailed bounds

on the convergence time. Those detailed bounds indicate that the convergence time only slightly improves with increasing k since it scales as $(1 + 1/k) \in [1, 2]$.

Our main result, the concentration bound, is stated in the second part of the theorem, with more detailed bounds derived in Proposition 5.8. These bounds do not depend on either the value of k or any structural properties of the graphs. Thus, for example, the variance of F on the clique and the cycle are asymptotically the same. The initial distribution of the values $\xi(0)$ also does not have impact, asymptotically. Note that if all initial values are $o(\sqrt{n})$, then $\mathbb{V}ar(F) = o(1)$, and by Chebychev's Inequality, with probability $1 - o(1)$, $F = o(1)$.

Our bounds on the convergence time are similar to the bounds on the convergence time of neighbourhood load balancing [11], which, in turn, can be related to the mixing time of random walks. Load-balancing (diffusion) bounds are often stated in terms of the *discrepancy* of $\xi(t)$ defined as $K = \max_i \xi_i(t) - \min_i \xi_i(t)$. Note that $\log(\|\xi(0)\|_2^2) \leq 2 \log(Kn)$, due to our assumption that $\sum_i \xi_i(0) = 0$. The additional factor of n in our convergence bound in comparison to [11] is due to the fact that we consider an asynchronous model where one node is activated at the time, while [11] considers a synchronous model.

Note that our convergence time bound does not give results for the voter model since we assume that α is a positive constant. For comparison, an upper bound of $O\left(n/(1 - \lambda_2(P))\right)$ on the expected convergence time of the *parallel* voter model for regular graphs is given in [18]. Thus, our process is faster by $\Omega(n/\log n)$, provided that K and $1/\varepsilon$ are polynomial in n.

DEFINITION 2.3 (*EdgeModel*). *At each time step $t \geq 1$, a directed edge $e = (u, v)$ is chosen uniformly among all edges. Then $\xi_u(t) = \alpha\xi_u(t-1) + (1-\alpha)\xi_v(t-1)$. For all $u' \neq u$, $\xi_{u'}(t+1) = \xi_{u'}(t)$.*

We obtain the following results for the *EdgeModel*. The proof of the results can be found in the full version. Let $\lambda_2(L)$ be the second-smallest eigenvalue of the Laplacian of G (defined in Section 4).

THEOREM 2.4 (*EdgeModel*). *Let G be a connected graph and α a constant in $(0, 1)$. Let $\varepsilon \in (0, 1)$, and let $T = T_\varepsilon$ be the first time such that $\xi(T)$ is ε-converged.*

(1) (Convergence time) Then, we have w.h.p.

$$T = O\left(\frac{m \log\left(n\|\xi(0)\|_2^2/\varepsilon\right)}{\lambda_2(L)}\right).$$

Furthermore, there exists an initial state $\xi(0)$ for which this bound is tight up to constants.

(2) (Concentration) If G is regular, we have

$$\mathbb{V}ar(F) = \Theta\left(\frac{\|\xi(0)\|_2^2}{n^2}\right)$$

It is interesting to note that the fixed point in the *EdgeModel* is, in expectation, the initial average—even for irregular graphs. The concentration follows from Theorem 2.2 since for regular graphs the *NodeModel* and the *EdgeModel* are identical (for $k = 1$). For d-regular graphs both theorems (Theorem 2.2 and Theorem 2.4)

give the same bound on the convergence time. Note that in this case there is a factor of d between $1 - \lambda_2(P)$ and $\lambda_2(L)$.

3 PREVIOUS RELATED RESULTS

Our protocols fit very well into the framework of opinion dynamics on graphs. The study of these dynamics has a long history in many academic disciplines, including physics, computer science, electrical engineering, population genetics and epidemiology. Processes falling into this category include the voter process [12, 18, 21], gossip algorithms [42], opinion dynamics [13, 23, 29, 34], consensus [5, 10, 20, 32], majority protocols [4, 19, 20, 25] diffusion load balancing [3, 38, 44], or so-called averaging processes [1].

Many of these models assume unilateral updates. For example, in the voter process, in each step a random agent adopts the opinion of a randomly chosen neighbour. Many majority protocols [19, 20] assume that an agent randomly chooses k of its neighbours and adopts the majority opinion. Our model is continuous version of this where we are allow to average between multiple opinions rather than limit the opinions to be a non-numeric set. In many opinion dynamics, e.g., [23, 29], the individuals communicate with all their neighbours before they adjust their opinions. These models are usually deterministic. The authors of [27] consider a randomised variant of the Friedkin-Johnsen model which is similar to our *NodeModel*. The agents are only allowed to interact with a small randomly chosen subset of their neighbours. The authors refer to their model as using only limited information. The motivation of the model is as follows. In today's social networks many users have a very large amount of followers/friends resulting in large degree nodes. In such a setting, it seems unnatural to assume that every agent/user communicates first with all its friends before making up their own mind about a subject (e.g., the esteem they hold for a brand). Instead, they might only communicate with a small subset of their friends before taking a decision.

Voting and Majority. Our model can be regarded as a generalisation of the voter model which was first analysed in [33]. In [33] the authors assume a network, and any node of the network has a discrete value. The authors show a bound of $O(t_m \cdot \log n) = O(n^3 \log n)$ where t_m is the expected meeting time of two random walks. In [18] the authors provide an improved upper bound of $O((1 - \lambda_2)^{-1} \log^4 n + \rho))$ on the expected consensus time for any graph G, where λ_2 is the second eigenvalue of the transition matrix of a random walk on G, with $\rho = \left(\sum_{u \in V(G)} d(u) \right)^2 / \sum_{u \in V(G)} d^2(u)$. The value of ρ ranges from $\Theta(1)$, for the star graph, to n, for regular graphs. The authors of [12] consider voting in dynamic graphs and analyse the consensus time in terms of the conductance ϕ of the underlying graphs. The authors show a bound on the expected consensus time of $O(d_{max} \cdot n/(d_{min} \cdot \phi))$ for graphs with minimum degree d_{min} and maximum degree d_{max}. A generalisation of this model for discrete opinions was introduced in [19]. Similar to our model, the nodes consider the opinion of k randomly chosen neighbours. In [19] the nodes adopt the majority opinion (the own opinion can be included or not), and not the average as in our models. There are several different models with different tie breaking-rules. See [5] for an overview article about related consensus dynamics.

Opinion Dynamics. There has been an interesting line of research trying to explain the spread of innovations and new technologies based on local interactions, where nodes are only allowed to communicate with their direct neighbours [16, 17, 26, 41]. The study of opinion-forming processes via local communication was introduced by DeGroot [23] where it is assumed that the underlying network is a directed graph with weighted edges. The nodes of the network are agents having opinions modelled as real numbers. The state of the network in step $t + 1$ is the product of the opinions in step t times the weighted adjacency matrix. If and how fast the process converges depends heavily on the matrix. For example, if the edge weights of node with degree d are chosen as $1/d$, the convergence time is roughly $\log n \cdot (1 - \lambda_{max})^{-1}$, where λ_{max} is the largest eigenvalue of the matrix (see [31]). In [48] the author uses a similar averaging approach to model the movement of particles. Friedkin and Johnsen (FJ) [29] extended the model by incorporating private opinions. n there, every agent has a private opinion, which does not change, and a so-called expressed opinion. The expressed opinion of an agent is defined as a deterministic function of the expressed opinions of all its neighbours and its private opinion. Another very influential model is the one by Hegselmann and Krause [34]. In this model the set of neighbours that influence a given agent is no longer fixed and the agents' opinions and their respective sets of influencing neighbours co-evolve over time. At any point in time the set of influencing neighbours of an agent consists of all the neighbours in a given static social network with an opinion close to their own opinion. In [28] the authors study convergence properties of a general model where the agents update their opinions in rounds to a weighted average of all opinions in their neighbourhoods. In [27] the authors consider a variant of the Friedkin Johnson model where the agents interact with a small subset of their neighbours. They refer to their model as using only limited information. The authors show convergence properties of simple and natural variants of the FJ in this setting.

Diffusion and Consensus. Another research area related to our work is that of diffusion protocols. Here a network of n identical nodes is given and every node stores a value. The value can, for example, model the load of the nodes or it can simply be a number. The protocol runs in parallel steps and in each step all nodes average their own value with the value of all neighbours. In the load balancing setting on a regular network with degree d, this is the same as sending a load of $\max\{0, \ell_u - \ell_v\}$ over the edge from u to v where ℓ_u (ℓ_v) is the load of node u (v). The objective is to distribute the load as evenly as possible among the nodes whilst minimizing the number of load balancing steps. If the values model numbers, diffusion can also be used to reach average consensus [38]. A variant of a diffusion process is the so-called dimension exchange process where the edges used for the balancing form a matching. The diffusion model was first studied by [22] and, independently, by [38]. The authors of [43] show a tight connection between the convergence rate of the diffusion algorithm and the absolute value of the second largest eigenvalue λ_{max} of the diffusion matrix P. P is defined as follows: $p_{ij} = 1/(d + 1)$ if $\{i, j\} \in E$ where d is the degree of the nodes. The convergence time is bounded by $2 \log(n^2 + K)/(1 - \lambda_{max})$, where K is the initial maximum load

difference. There is a similar connection between $\lambda_{\max}$ and the convergence time for the dimension exchange model [11].

There is a vast amount of literature about diffusion processes in different research communities. In [47] the authors observe relations between convergence time and properties of the underlying network, like electrical and fluid conductance. In [38] there is a nice overview about the results. The author distinguishes between consensus (all nodes have to agree on a value) or average consensus where nodes have to calculate the average, similar to neighbourhood load balancing. The main goal of the author is to develop diffusion-type algorithms where the nodes are only aware of their own edges, but neither of the edges of their neighbours nor of their degrees. The author presents algorithms for both types of consensus with convergence time $\widetilde{O}(n^4)$ and dynamic networks, together with randomised algorithms for undirected graphs.

Many publications focus on average consensus via diffusion type processes. In [35] the authors present a gossip-based algorithm (which is similar to an averaging algorithm) for complete graphs. The authors show that the protocol can be used for the computation of sums, averages, random samples, quantiles, and other aggregate functions. They also show that our protocols converge exponentially fast. In [6, 14] the authors generalise the result to arbitrary graphs. In [14] the authors show that the averaging time of their algorithm depends on the second largest eigenvalue of a doubly stochastic matrix. [24, 46] consider averaging algorithms for directed graphs where agents transmit their values to one or several agents, but they do not receive data. Nodes update their values using a weighted linear combination of their own value and the values of neighbouring nodes. In [3] the authors study a dynamic load balancing process on cycles. In [15] assume that the load inputs are drawn from a fixed probability distribution. In [36, 37, 40] the authors use a diffusion algorithm for counting the nodes in an anonymous network and in [25] diffusion is used for a majority process in the population model. In [30, 45] the authors compare continuous with discrete diffusion processes.

4 NOTATION AND PRELIMINARIES

In order to prove detailed concentration bounds (stated in Proposition 5.8), we introduce additional notation and new concepts. We denote our Averaging Process with $(\xi(t))_{t \geq 0}$, where $\xi(t) = (\xi_u(t))_{u \in V}$. For any vector $\xi(t) \in \mathbb{R}^n$, we define the quantities

$$\text{Avg}(t) = \frac{1}{n} \sum_{u \in V} \xi_u(t) \qquad \text{and} \qquad M(t) = \sum_{u \in V} \frac{d_u}{2m} \cdot \xi_u(t). \quad (1)$$

For any vector $v \in \mathbb{R}^n$, let $\|v\|_2 = \sqrt{\sum_i v_i^2}$. Let $L = D - A$ be the graph Laplacian of G, where $D = \text{diag}(d_1, ..., d_n)$ is the diagonal matrix with entries d_i (degrees of nodes), and A is the adjacency matrix of G. L is a symmetric positive semi-definite matrix with eigenvalues $0 = \lambda_1(L) < \lambda_2(L) \leq \cdots \leq \lambda_n(L)$. Note that $\lambda_2(L) > 0$ follows from the connectivity of G (L is irreducible). Further, $e^{(i)} \in \{0, 1\}^n$ is the indicator (column) vector, where the i-th entry is 1 and all other entries are 0, $\mathbf{1}$ is a vector of all 1's (of appropriate length to fit the context), and symbol $\top$ indicates the transpose of a vector or matrix (introduced to avoid confusion with the time step T).

We consider the following *lazy variant* of the *NodeModel*, where in each step with probability $1/2$ the selected node performs no

update, i.e., $\xi(t) = \xi(t - 1)$ and otherwise behaves as before. This can be related to the transition matrix P of a lazy random walk on G, with $p_{(i,i)} = 1/2$ and $p_{(i,j)} = 1/(2d_i)$, for each $i \in V$ and $(i, j) \in E$. Let $\lambda_2(P)$ be the second-largest eigenvalue of P and let $f_2(P)$ be the corresponding eigenvector. Let π denote the vector with π_i equal to the probability that a fully mixed random walk is at the node i. We will use the π weighted inner product of two vectors v and v' in $\mathbb{R}^n$ given by

$$\langle v, v' \rangle_\pi = \sum_{x \in V} \pi_x v_x v'_x. \quad (2)$$

Let $\pi_{max} = d_{\max}/2m$ and $\pi_{min} = d_{\min}/2m$. Our goal is to analyse the time it takes until the nodes have almost identical values. To do so, we will make use of the following potential function.

$$\phi(\xi(t)) = \langle \xi(t), \xi(t) \rangle_\pi - \langle \mathbf{1}, \xi(t) \rangle_\pi^2$$
$$= \frac{1}{2} \sum_{u,v \in V} \pi_u \pi_v (\xi_u(t) - \xi_v(t))^2. \quad (3)$$

We say that the process has ε-*converged* whenever $\phi(\xi(t)) \leq \varepsilon$. For comparison with related notions: a $(\varepsilon/n)^6$-convergence implies a discrepancy K (maximum value minus minimum value) of at most ε. Finally, a fundamental property of the processes defined in Eq. (1) is that they are martingales, as shown in the following result (its proof is in the full version).

Lemma 4.1. *We have $\mathbb{E}(M(t + 1)|\xi(t)) = M(t)$. In particular, $\mathbb{E}(M(t+1)|\xi(0)) = M(0)$ and for regular graphs $\mathbb{E}(M(t+1)|\xi(0)) = \text{Avg}(0)$.*

5 PROOF OF THE CONCENTRATION BOUNDS

In this section we will show the second part of our main theorem which states results for the *NodeModel*. Note that this part of the theorem holds for regular graphs only, and recall that the results extend to the *EdgeModel*, since for regular graphs both models are identical. To show our result we will study the dual $(W(t))_{t \geq 0}$ of the Averaging Process $(\xi(t))_{\geq t}$, which we call the Diffusion Process (for a detailed definition of the process see Section 5.1). The latter process can be thought of as an n-dimensional diffusion process in which the dimensions are balanced separately but not independently of each other. Due to the duality (which we show in Proposition 5.1) we get that the variance of the final node values of both processes is the same. Therefore, it is sufficient to study the variance of the dual process. In turn, to analyse the variance of the dual process we study the joint distribution μ of two Random Walk Processes $(\widetilde{W}(t))_{t \geq 0}$ (see Section 5.2 for a detailed definition), where two random walks start in two (not necessarily different) nodes. We will present and prove the following relationships.

$$\mathbb{V}ar(M(t)) \underset{\text{Lemma 5.2}}{\approx} \mathbb{V}ar(W(t)) \underset{\text{Proposition 5.4}}{\approx} \mathbb{V}ar(\widetilde{W}(t))$$
$$\underset{\text{Lemma 5.5}}{\approx} \sum_{u,v} \mu(u, v) \xi_u(0) \xi_v(0)$$

Through Lemma 5.2 and Proposition 5.4 we show that the three processes have the same variance. Then, through Lemma 5.5 we will show that this variance can be expressed in terms of μ and $\xi(0)$. Recall that we use $\top$ to indicate the transpose of a vector.

5.1 The Diffusion Process

We introduce the Diffusion Process, which is closely related to our Averaging Process. We call this relationship *duality*. The Diffusion Process takes as parameters a cost (row) vector $c \in \mathbb{R}^n$ and an initial load (column) vector $q(0) \in \mathbb{R}^n$. At a time step $t \geq 0$, the load vector is denoted by $q(t) \in \mathbb{R}^n$ and its cost is the value $c\,q(t)$. At every time step $t \geq 1$, a node $u(t)$ is sampled uniformly at random. Node $u(t)$ then chooses a subset $S(t)$ of k neighbours uniformly at random and spreads a $1 - \alpha$ fraction of its load uniformly to these k neighbours. The choice of neighbours and load redistribution can be represented using matrix $B(t)$ defined as follows.

$$B_{i,j}(t) = \begin{cases} 1, & \text{if } i = j \neq u(t), \\ \alpha, & \text{if } i = j = u(t), \\ (1-\alpha)/k, & \text{if } i \in S(t) \text{ and } j = u(t), \\ 0, & \text{otherwise.} \end{cases} \qquad (4)$$

Then at step $t \geq 1$, the load vector and the state of the process (the cost of the load distribution) are given, respectively, by

$$q(t) = B(t)\,q(t-1) = R(t)\,q(0) \quad \text{and} \quad w(t) = c\,q(t) = c\,R(t)\,q(0),$$

where

$$R(t) = B(t) \cdot B(t-1) \cdots B(1). \qquad (5)$$

The Diffusion Process can be applied simultaneously to a number of load vectors $q^{(1)}, q^{(2)}, \ldots, q^{(r)}$, $r \geq 1$, of different commodities:

$$(q^{(1)}(t), q^{(2)}(t), \ldots, q^{(r)}(t))$$
$$= R(t)\,(q^{(1)}(0), q^{(2)}(0), \ldots, q^{(r)}(0)),$$
$$W(t) = c\,R(t)\,(q^{(1)}(0), q^{(2)}(0), \ldots, q^{(r)}(0)),$$

where $q^{(j)}(t)$ and $W^{(j)}(t)$ (j-th entry in $W(t)$) are, respectively, the load vector j and its cost at step t. Note that $W(t) \in \mathbb{R}^r$. Next, we formally state the duality between the Averaging Process and the Diffusion Process. For $u \in V$, the column vector $e^{(u)}$ is the unit vector with 1 at position u and 0's everywhere else.

Proposition 5.1. *If the Diffusion Process is applied to cost vector $c = \xi^\top(0)$ and to n initial load vectors $e^{(u)}$, $u \in V$ (that is, the total load – one unit – of 'commodity' u is initially at node u), then for each $T \geq 0$, the probability distribution of $\xi(T)$ in the Averaging Process is the same as the probability distribution of $W(T)$ in the Diffusion Process. That is, for any (column) vector $a \in \mathbb{R}^n$, $\mathbb{P}(\xi(T) = a) = \mathbb{P}(W(T) = a^\top)$.*

Proof. This duality relation follows by coupling the Averaging Process with the Diffusion Process, running one of them (say the Averaging Process) forward in time and the other backwards in time. More formally, we fix an arbitrary time step $T \geq 1$ and consider any feasible node selection sequence $\mathcal{X} = (\mathcal{X}(1), \mathcal{X}(2), \ldots, \mathcal{X}(T))$ for T steps. That is, for $1 \leq t \leq T$, $\mathcal{X}(t) = (S(t), u(t))$, where $u(t)$ and $S(t)$ are a node and a size k sample of its neighbours. The lemma below shows that if we run the Averaging Process using sequence $\mathcal{X}$ and the Diffusion Process using the reverse sequence $\mathcal{X}^R$, then $W(T) = \xi^\top(T)$. The proposition then follows because the probability of having sequences $\mathcal{X}$ in the first T steps in the Averaging Process is the same as the probability of having sequence $\mathcal{X}^R$ in the first T steps in the Diffusion Process.

Note that running one of the two processes backward is crucial for establishing this duality. If both processes are run forward on the same sequence $\mathcal{X}$, then most likely $W(T) \neq \xi^\top(T)$. □

Lemma 5.2. *For the Averaging Process and the Diffusion Process as in Proposition 5.1, an arbitrary time step $T \geq 1$, and an arbitrary node selection sequence $\mathcal{X} = (\mathcal{X}(1), \mathcal{X}(2), \ldots, \mathcal{X}(T))$, if we run the Averaging Process using sequence $\mathcal{X}$ and the Diffusion Process using the reverse sequence $\mathcal{X}^R$, then we have $W(T) = \xi^\top(T)$.*

Proof. Since for each $1 \leq t \leq T$, in step t the Averaging Process uses the same node selection as in the Diffusion Process in step $T + 1 - t$. Then $\xi(t) = F(t)\xi(t-1)$, where $F(t) = B^\top(T - t)$ and thus we have

$$\begin{aligned} W(T) &= c \cdot (B(T)B(T-1)\ldots B(1)) \cdot \left(e^{(1)}, e^{(2)}, \ldots, e^{(n)}\right) \\ &= c \cdot (B(T)B(T-1)\ldots B(1)) \cdot I \\ &= \xi^\top(0) \cdot F^\top(1)F^\top(2)\ldots F^\top(T) \\ &= (F(T)\ldots F(2)F(1) \cdot \xi(0))^\top = \xi^\top(T). \end{aligned}$$

□

From now on, Diffusion Process means the diffusion process as specified in Proposition 5.1. For an illustrative example of how the Averaging Process and the Diffusion Process work and relate to each other, see Fig. 1. In there, we fix $\alpha = 1/2$ and $k = 1$. For another example with $\alpha = 1/2$ and $k > 1$ see the full version.

5.2 The Random Walk Process

With the Diffusion Process we associate n random walks $\left(\tilde{q}^{(u)}(t)\right)_{t \geq 0}$, for $u \in V$. At step t, the position of walk u is $\tilde{q}^{(u)}(t) \in \{e^{(1)}, e^{(2)}, \ldots, e^{(n)}\}$, where the '1' in vector $\tilde{q}^{(u)}(t)$ indicates the node where the walk is at this step. The cost of this walk at step t is defined as $\widetilde{W}^{(u)}(t) = \xi(0) \cdot \tilde{q}^{(u)}(t)$, from the Averaging Process. Thus, if the walk is at a node $v \in V$, its cost is $\widetilde{W}^{(u)}(t) = \xi_v(0)$. The starting node of this walk is node u, that is, $\tilde{q}^{(u)}(0) = e^{(u)}$. At step t, the transition matrix for all n random walks is matrix $B(t)$ of the Diffusion Process given in Eq. (4). That is, given the location of random walk u at step $t - 1$, represented by vector $\tilde{q}^{(u)}(t-1)$, and fixing the transition matrix $B(t)$, the distribution of the location of this walk at step t is given by $B(t)\tilde{q}^{(u)}(t-1) = R(t)e^{(u)}$. Hence, if $B(t)$ represents choosing node v and its k neighbours $w_1, w_2, \ldots, w_k$, and if the walk happens to be at node v at step $t - 1$, then in step t the walk moves to node w_i with probability $(1 - \alpha)/k$, for $i = 1, 2, \ldots, k$. Otherwise, the walk does not move in step t. In what follows, 'the random walk starting at node u' means the random walk defined here. The relation between the random walks defined above and the Diffusion Process is akin to the relation between the standard random walk and the standard diffusion process.[3] The former describes the location of the random walk at a given time (the walk can only be at one position at any time), while the latter gives the distribution of the random walk at any given time step.

---

[3] In the standard random walk process the walk moves to all neighbors with equal probability and in the standard diffusion process, each node sends the same fraction of load to all its neighbors synchronously starting with load one at one node and load zero elsewhere.

(a)

(b)
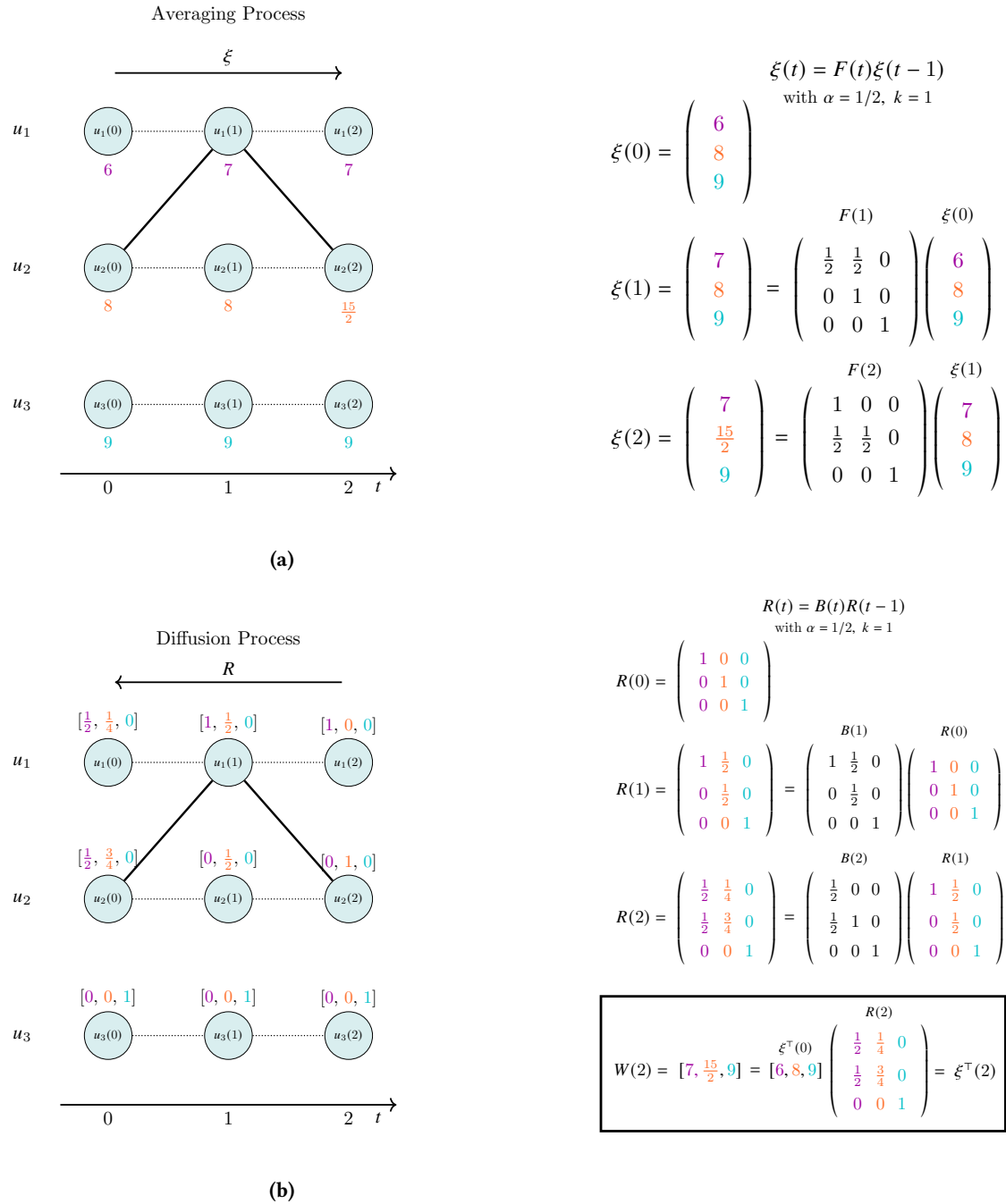
Figure 1: Illustration of the duality between the Averaging Process (a) and the Diffusion Process (b), with $k = 1$ and $\alpha = 1/2$. In (a), at $t = 1$ node u_1 and its neighbour u_2 are selected, the value at u_1 is updated and the values at u_2 and u_3 stay the same – see matrix $F(1)$. At $t = 2$, u_2 and its neighbour u_1 are selected, leading to $\xi(2)$ – see matrix $F(2)$. In (b), the Diffusion Process runs backwards. The initial state, at $t = 2$, of the diffusion starting in u_2 is the vector $[0, 1, 0]$. At the first step ($t = 2$), u_2 sends $1 - \alpha = 1/2$ of its load to u_1. The loads in the other nodes do not spread. The resulting load vector is $R_2(1) = [1/2, 1/2, 0]$, the second column of $R(1)$. After the second step, the load is $R_2(2) = [1/4, 3/4, 0]$. The diffusion of the loads originating at nodes u_1 and u_3 is indicated in purple and cyan respectively. We get $W(2) = \xi^{\top}(2)$.

LEMMA 5.3. *For any fixed sequence of transition metrices* $\chi = (B(1), B(2), \ldots, B(t))$, *for the random walk starting at node u, we have (recall the definition if $R(t)$ in (5)),*

$$\mathbb{E}[\tilde{q}^{(u)}(t) \mid \chi] = R(t)e^{(u)}, \tag{6}$$

$$\mathbb{E}[\widetilde{W}^{(u)}(t) \mid \chi] = W^{(u)}(t), \tag{7}$$

PROOF. Given χ, the vector $R(t)e^{(u)}$ is the distribution at step t of the random walk starting at node u. This means that for $x \in V$, the expectation of $\tilde{q}_x^{(u)}(t) \in \{0, 1\}$ is equal to entry x in $R(t)e^{(u)}$, which gives the probability that at step t the walk is at node x (the probability that $\tilde{q}_x^{(u)}(t) = 1$). Hence Eq. (6).[4] For Eq. (7), by linearity of expectation,

$$\begin{aligned}
\mathbb{E}[\widetilde{W}^{(u)}(t) \mid \chi] &= \mathbb{E}[\xi^\top \tilde{q}^{(u)}(t) \mid \chi] \\
&= \xi^\top \cdot \mathbb{E}[\tilde{q}^{(u)}(t) \mid \chi] = \xi^\top R(t) e^{(u)} \\
&= W(t)e^{(u)} = W^{(u)}(t).
\end{aligned}$$

where $\xi = \xi(0)$. □

We refer collectively to the n random walks defined in this section as Random Walk Process $\widetilde{W}(t)$. These n random walks are correlated since they use the same random choices of nodes and neighbours sampled, that is, the same transition matrices $B(t)$.

PROPOSITION 5.4. *For the Diffusion Process $W(t)$ and the Random Walk Process $\widetilde{W}(t)$, we have*

$$\mathbb{E}[\widetilde{W}^{(u)}(t)\widetilde{W}^{(v)}(t)] = \mathbb{E}[W^{(u)}(t)W^{(v)}(t)] \tag{8}$$

PROOF. It is sufficient to prove that for any fixed sequence χ of transition matrices,

$$\mathbb{E}[\widetilde{W}^{(u)}(t)\widetilde{W}^{(v)}(t) \mid \chi] = W^{(u)}(t)W^{(v)}(t). \tag{9}$$

Note that if χ is fixed, then there is no randomness on the right-hand side of the above. By writing ξ instead of $\xi(0)$, we have

$$\begin{aligned}
\mathbb{E}[\widetilde{W}^{(u)}(t)\widetilde{W}^{(v)}(t) \mid \chi] &= \mathbb{E}[\xi^\top \tilde{q}^{(u)}(t) \cdot \tilde{q}^{(v)\top}(t)\xi \mid \chi] \\
&= \xi^\top \cdot \mathbb{E}[\tilde{q}^{(u)}(t) \cdot \tilde{q}^{(v)\top}(t) \mid \chi] \cdot \xi \tag{10} \\
&= \xi^\top \cdot \mathbb{E}[\tilde{q}^{(u)}(t) \mid \chi] \cdot \mathbb{E}[\tilde{q}^{(v)\top}(t) \mid \chi] \cdot \xi \tag{11} \\
&= \left(\xi^\top R(t)e^{(u)}\right) \cdot \left(e^{(v)^\top} R^\top(t)\xi\right) \tag{12} \\
&= W^{(u)}(t) \cdot W^{(v)}(t),
\end{aligned}$$

where (10) is by linearity of expectation, (11) is by independence of the random walks $\tilde{q}^{(u)}(t)$ and $\tilde{q}^{(v)}(t)$ (once the sequence χ is fixed), and (12) follows from Eq. (6). □

---

[4] $\mathbb{E}[(X_1, X_2, \ldots, X_n)]$ is defined as $(\mathbb{E}[X_1], \mathbb{E}[X_2], \ldots, \mathbb{E}[X_n])$.

5.3 Joint Distribution of Two Random Walks

In this section we consider two correlated random walks $\tilde{q}^{(a)}(t)$ and $\tilde{q}^{(b)}(t)$, $a, b \in V$, $a \neq b$. Respectively, we denote by $X(t)$ and $Y(t)$ the nodes where these walks are at step t. The random walks proceed through the Random Walk Process described in the previous section, resulting in a joint transition matrix Q:

$$\begin{aligned}
&Q((x, y), (u, v)) \\
&= \mathbb{P}((X(t+1), Y(t+1)) = (u, v) \mid (X(t), Y(t)) = (x, y)),
\end{aligned}$$

where $(x, y), (u, v) \in V \times V$. The Q chain defined by this transition matrix is irreducible (each state (u, v) is reachable form each state (x, y)) and aperiodic ($Q(s, s) > 0$, for each $s \in V \times V$), so it has a unique stationary distribution. We use $\pm f(n)$ to denote a term $cf(n)$ where $|c| \leq 1$.

LEMMA 5.5. *For the stationary distribution μ of the Q chain defined above and sufficiently large T, we have,*

$$\mathbb{E}[\widetilde{W}^{(a)}(T)\widetilde{W}^{(b)}(T)] = \sum_{u,v} \mu(u, v)\xi_u(0)\xi_v(0) \pm \frac{1}{n^5}. \tag{13}$$

PROOF. Let T be the mixing time of Q such that $\left|\mu(u, v) - Q^T((a, b), (u, v))\right| \leq \frac{1}{K^2 n^7}$, for each $(u, v) \in V \times V$, where K is the initial discrepancy. Now, let's write ξ instead of $\xi(0)$, then

$$\begin{aligned}
\mathbb{E}\left[\widetilde{W}^{(a)}(T)\ \widetilde{W}^{(b)}(T)\right] \\
\overset{(*)}{=} \mathbb{E}\left[\xi^\top \tilde{q}^{(a)}(T) \left(\tilde{q}^{(b)}\right)^\top (T)\, \xi\right] \\
= \sum_{u,v} \Pr\left(\tilde{q}^{(a)}(T) = e^{(u)}, \tilde{q}^{(b)}(T) = e^{(v)}\right)\xi_u\xi_v \\
= \sum_{u,v} Q^T((a, b), (u, v))\xi_u\xi_v \\
= \sum_{u,v} \left(\mu(u, v) \pm \frac{1}{K^2 n^7}\right)\xi_u\xi_v \\
= \sum_{u,v} \mu(u, v)\xi_u\xi_v \pm \frac{1}{n^5},
\end{aligned}$$

where (*) follows from the definition of Random Walk Process. □

We calculate the entries of matrix Q and find its exact stationary distribution. From our setting, there are 3 different types of transitions: neither walk leaves its current node (a self-loop), only one walk moves, or both walks move. If the walks are in the same node, then they can both travel to the same node or to two different nodes (or one or both could stay in their current node). The transition where both walks are moving requires that they be in the same node.

Case1: Both walks are at the same node x. Then, for some nodes u and v s.t. $x \neq u \neq v \neq x$,

$$Q((x, x), (u, v)) = (1-\alpha)^2 \pi_x \frac{k}{d}\frac{k-1}{d-1}\frac{1}{k^2} = (1-\alpha)^2 \pi_x \frac{k-1}{kd(d-1)}, \tag{14}$$

$$Q((x, x), (u, u)) = (1-\alpha)^2 \pi_x \frac{k}{d}\frac{1}{k^2} = (1-\alpha)^2 \pi_x \frac{1}{kd}, \tag{15}$$

$$Q((x, x), (x, u)) = \alpha(1 - \alpha)\pi_x \frac{1}{d}, \tag{16}$$

$$Q((x, x), (u, x)) = \alpha(1 - \alpha)\pi_x \frac{1}{d}, \tag{17}$$

$$Q((x, x), (x, x)) = \alpha^2 \pi_x + (1 - \pi_x). \tag{18}$$

We explain the meaning of Eq. (14), the other equations follow in a similar manner. If both walks are in x, then for them to have a chance of moving, we need to sample node x first. This happens with probability π_x. The probability of both walks moving away from x is equal to $(1 - \alpha)^2$. The first walk goes to u and the second to v, if both u and v are in the selected k-sample of neighbours of x – probability $\frac{k(k-1)}{d(d-1)}$ – and then the first walk chooses u and the second one chooses v – probability $\frac{1}{k^2}$.

Case2: The walks are on two different nodes $x \neq y$. Then, for a node $v \neq y$ and a node $u \neq x$,

$$Q((x, y), (x, v)) = (1 - \alpha)\pi_y \frac{1}{d} \tag{19}$$

$$Q((x, y), (u, y)) = (1 - \alpha)\pi_x \frac{1}{d} \tag{20}$$

$$Q((x, y), (x, y)) = (1 - \pi_x - \pi_y) + (\pi_x + \pi_y)\alpha \tag{21}$$

All other transition probabilities are 0. The cases above hold for any graph. In the case of a d-regular graph, then we have that $P(x, u) = \frac{1}{d} \; \forall u \in N(x)$. Note that Q is not reversible, however a stationary distribution is still possible to find, since the chain is irreducible and positive recurrent. For an example explaining why Q is not reversible, see the proof of Lemma 5.7.

Next, we define three sets of different types of states of the Q-chain and give the formula for the stationary distribution of this chain.

Definition 5.6. *Consider a d-regular graph $G = (V, E)$ on n nodes, and for $u, v \in V$, let $dis(u, v)$ denote the length of a shortest path between u and v in G. For $i \geq 0$, we define sets $S_i = \{(u, v) \mid u, v \in V, dis(u, v) = i\}$ as subsets of the state space $V \times V$ of the Q-chain. We also define $S_+ = \bigcup_{i \geq 2} S_i$.*

Lemma 5.7. *Consider the NodeModel with parameters $\alpha \in (0, 1)$ and $k \geq 1$. The stationary distribution μ of the Q-chain is comprised of exactly three different values: for a state $(u, v) \in V \times V$,*

$$\mu(u, v) = \begin{cases} \mu_0(n, d, k, \alpha) = 2k(d-1)\ell & \text{if } (u, v) \in S_0 \\ \mu_1(n, d, k, \alpha) = (d-1)\gamma\ell & \text{if } (u, v) \in S_1 \\ \mu_+(n, d, k, \alpha) = (d\gamma - 2\alpha k)\ell & \text{if } (u, v) \in S_+ \end{cases} \tag{22}$$

with $\gamma = k(1 + \alpha) - (1 - \alpha)$, and $\ell = \frac{1}{n(n(d\gamma - 2\alpha k) + 2(1-\alpha)(d-k))}$.

All missing proofs can be found in the full version.

5.4 Proof of Theorem 2.2(2): Concentration of the Convergence Value

We present here the following proposition, which we use to show the second part of Theorem 2.2.

Proposition 5.8. *Consider NodeModel with parameters $\alpha \in (0, 1)$ and $k \geq 1$. W.l.o.g. we assume that $Avg(0) = 0$. Let $E^+ = \{(u, v)|\{u, v\} \in E\}$ be the set of directed edges in the underlying graph. Then, for any $t \geq 0$,*

$$\mathbb{V}ar(Avg(t)) \leq$$

$$\leq (\mu_0 - \mu_+) \sum_{u \in V} \xi_u^2(0) + (\mu_1 - \mu_+) \sum_{(u,v) \in E^+} \xi_u(0)\xi_v(0) + 1/n^5, \tag{23}$$

Furthermore, there exist a T large enough such that for all $t \geq T$

$$\mathbb{V}ar(Avg(t)) \geq$$

$$\geq (\mu_0 - \mu_+) \sum_{u \in V} \xi_u^2(0) + (\mu_1 - \mu_+) \sum_{(u,v) \in E^+} \xi_u(0)\xi_v(0) - 1/n^5, \tag{24}$$

Proof. The random variable $Avg(t) = \frac{1}{n} \sum_{x \in V} \xi_x(t)$ is a Martingale (Lemma 4.1). Due to our convergence results, we know that as $t \to \infty$, all node values $\xi_x(t)$ converge to the same value. Consequently, $Avg(t)$ converges to the same value, which we denote by $Avg(\infty)$. This is a random variable with expectation $\mathbb{E}[Avg(\infty)] = Avg(0)$, since by the Martingale property, for each $t \geq 0$, $\mathbb{E}[Avg(t)] = Avg(0)$. We want to show that the actual value of $Avg(\infty)$ is likely to be close to $Avg(0)$. We do this by showing that $\mathbb{V}ar(Avg(\infty)) = \lim_{t \to \infty} \mathbb{V}ar(Avg(t))$ is small. We start by recalling the assumption that $Avg(0) = 0$ and using linearity of expectation to obtain

$$\mathbb{V}ar(Avg(t)) = \mathbb{E}[Avg(t)^2] - (\mathbb{E}[Avg(t)])^2$$
$$= \mathbb{E}[Avg(t)^2]$$
$$= \frac{1}{n^2} \sum_{x, y \in V} \mathbb{E}[\xi_x(t)\xi_y(t)].$$

Fixing $t = T$ arbitrarily, Propositions 5.1 and 5.4 imply that for any pair of nodes x and y, the three products $\xi_x(T)\xi_y(T)$, $W^{(x)}(T)W^{(y)}(T)$ and $\widetilde{W}^{(x)}(T)\widetilde{W}^{(y)}(T)$ have the same expectation, so

$$\mathbb{V}ar(Avg(T)) = \frac{1}{n^2} \sum_{x, y} \mathbb{E}[\xi_x(T)\xi_y(T)]$$
$$= \frac{1}{n^2} \sum_{x, y} \mathbb{E}[W^{(x)}(T)W^{(y)}(T)]$$
$$= \frac{1}{n^2} \sum_{x, y} \mathbb{E}[\widetilde{W}^{(x)}(T)\widetilde{W}^{(y)}(T)].$$

Let $\xi = \xi(0)$, then by Lemma 5.5 and Lemma 5.7 we get

$$\mathbb{V}ar(Avg(T)) = \frac{1}{n^2} \sum_{x, y} \left(\sum_{u, v} \mu(u, v)\xi_u\xi_v \pm 1/n^5 \right) = \sum_{u, v} \mu(u, v)\xi_u\xi_v \pm 1/n^5$$

$$= \mu_0 \sum_{(u,u) \in S_0} \xi_u^2 + \mu_1 \sum_{(u,v) \in S_1} \xi_u\xi_v + \mu_+ \sum_{(u,v) \in S_+} \xi_u\xi_v \pm 1/n^5$$

$$= (\mu_0 - \mu_+) \sum_{(u,u) \in S_0} \xi_u^2 + (\mu_1 - \mu_+) \sum_{(u,v) \in S_1} \xi_u\xi_v +$$

$$+ \mu_+ \sum_{u,v \in V} \xi_u\xi_v \pm 1/n^5$$

$$= (\mu_0 - \mu_+) \sum_{u \in V} \xi_u^2 + (\mu_1 - \mu_+) \sum_{(u,v) \in E} \xi_u^2 \pm 1/n^5,$$

using for the last equation that $\sum_{u,v} \xi_u\xi_v = (\sum_u \xi_u)(\sum_u \xi_u) = 0$ (the assumption that $Avg(0) = 0$). Thus Eq. (24) holds. For Eq. (24), observe that $\mathbb{V}ar(Avg(t))$ is non-decreasing. □

Note that our techniques only allow us to derive bounds on the variance that come from the mixing time of the Q-chain. Hence, we do not have tight bounds on the variance at the beginning of the process. However, at the end of the process (after the mixing time) our results are asymptotically tight. We are now ready to prove Theorem 2.2(2).

PROOF. Let $\xi = \xi(0)$, and, referring to the right-hand sides in (23) and (24), we derive

$$
(\mu_0 - \mu_+) \sum_{u \in V} \xi_u^2 \; + \; (\mu_1 - \mu_+) \sum_{(u,v) \in E^+} \xi_u \xi_v =
$$
$$
= ((\mu_0 - \mu_+) - d(\mu_1 - \mu_+)) \|\xi\|_2^2
$$
$$
+ (\mu_1 - \mu_+) \left(\sum_{(u,v) \in E^+} \xi_u \xi_v + d \sum_{u \in V} \xi_u^2 \right) \quad (25)
$$

Observe that $0 \le \sum_{(u,v) \in E^+} \xi_u \xi_v + d \sum_{u \in V} \xi_u^2 \le 2d\|\xi\|_2^2$, which follows from $\sum_{(u,v) \in E^+} \xi_u \xi_v + d \sum_{u \in V} \xi_u^2 = \sum_{\{u,v\} \in E} (\xi_u + \xi_v)^2 \le \sum_{\{u,v\} \in E} 2(\xi_u^2 + \xi_v^2) = 2d \sum_{u \in V} \xi_u^2 = 2d\|\xi\|_2^2$. Then, by noting that $\mu_1 - \mu_+ \le 0$, we have from Eq. (23) and Eq. (25) that $\mathbb{V}ar(F) - 1/n^5 \le ((\mu_0 - \mu_+) - d(\mu_1 - \mu_+))\|\xi\|_2^2 = \frac{2k(d-1)(1-\alpha)}{n^2(3dk+d-3k)}\|\xi\|_2^2 = O\left(\frac{\|\xi\|_2^2}{n^2}\right)$. From Eq. (24) and Eq. (25) we obtain $\mathbb{V}ar(F) + 1/n^5 \ge \frac{2k(d-1)(1-\alpha)}{n^2(3dk+d-3k)}\|\xi\|_2^2 + \frac{-k(1+\alpha)+(1-\alpha)+2\alpha k}{n^2(3dk+d-3k)}2d\|\xi\|_2^2 = \frac{2(1-\alpha)(2dk-d-k)}{n^2(3dk+d-3k)}\|\xi\|_2^2 = \Omega\left(\frac{\|\xi\|_2^2}{n^2}\right)$. □

6 FUTURE WORK

There are two enticing lines of future work. First, we have shown how to obtain tight bounds on $\mathbb{V}ar(F)$ by analysing the distribution of two dependent random walks. Can one obtain bounds on higher moments $M > 2$ by considering M-dependent random walks? This would allow to derive stronger Chernoff-type results for the concentration of F. Second, is it possible to bound the concentration in the *NodeModel* and *EdgeModel* for irregular graphs?

ACKNOWLEDGMENTS

REFERENCES

[1] David Aldous. 2013. Interacting particle systems as stochastic social dynamics. *Bernoulli* 19, 4 (2013), 1122–1149.

[2] David Aldous and Daniel Lanoue. 2012. A lecture on the averaging process. *Probability Surveys* 9, none (2012), 90 – 102. https://doi.org/10.1214/11-PS184

[3] Dan Alistarh, Giorgi Nadiradze, and Amirmojtaba Sabour. 2020. Dynamic Averaging Load Balancing on Cycles. In *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 168)*, Artur Czumaj, Anuj Dawar, and Emanuela Merelli (Eds.). Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 7:1–7:16. https://doi.org/10.4230/LIPIcs.ICALP.2020.7

[4] Dana Angluin, James Aspnes, and David Eisenstat. 2008. A simple population protocol for fast robust approximate majority. *Distributed Computing* 21, 2 (2008), 87–102. https://doi.org/10.1007/s00446-008-0059-z

[5] Luca Becchetti, Andrea E. F. Clementi, and Emanuele Natale. 2020. Consensus Dynamics: An Overview. *SIGACT News* 51, 1 (2020), 58–104. https://doi.org/10.1145/3388392.3388403

[6] Florence Bénézit, Vincent D. Blondel, Patrick Thiran, John N. Tsitsiklis, and Martin Vetterli. 2010. Weighted Gossip: Distributed Averaging using non-doubly stochastic matrices. In *IEEE International Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Proceedings*. IEEE, Austin, Texas, USA, 1753–1757. https://doi.org/10.1109/ISIT.2010.5513273

[7] Gerardo Beni and Ping Liang. 1996. Pattern reconfiguration in swarms-convergence of a distributed asynchronous and bounded iterative algorithm. *IEEE Transactions on Robotics and Automation* 12, 3 (1996), 485–490.

[8] Petra Berenbrink, Andrea Clementi, Robert Elsässer, Peter Kling, Frederik Mallmann-Trenn, and Emanuele Natale. 2017. Ignore or comply? On breaking symmetry in consensus. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*. ACM, Washington, DC, 335–344.

[9] Petra Berenbrink, Colin Cooper, Cristina Gava, David Kohan Marzagão, Frederik Mallmann-Trenn, Nicolás Rivera, and Tomasz Radzik. arXiv:2211.17125, 2023. Distributed Averaging in Opinion Dynamics. arXiv:2211.17125 [math.PR]

[10] Petra Berenbrink, Tom Friedetzky, George Giakkoupis, and Peter Kling. 2016. Efficient Plurality Consensus, Or: the Benefits of Cleaning up from Time to Time. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016 (LIPIcs, Vol. 55)*, Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi (Eds.). ICALP, Rome, Italy, 136:1–136:14. https://doi.org/10.4230/LIPIcs.ICALP.2016.136

[11] Petra Berenbrink, Tom Friedetzky, and Zengjian Hu. 2009. A new analytical method for parallel, diffusion-type load balancing. *J. Parallel Distributed Comput.* 69, 1 (2009), 54–61. https://doi.org/10.1016/j.jpdc.2008.05.005

[12] Petra Berenbrink, George Giakkoupis, Anne-Marie Kermarrec, and Frederik Mallmann-Trenn. 2016. Bounds on the Voter Model in Dynamic Networks. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy (LIPIcs, Vol. 55)*, Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Rome, Italy, 146:1–146:15. https://doi.org/10.4230/LIPIcs.ICALP.2016.146

[13] Arnab Bhattacharyya, Mark Braverman, Bernard Chazelle, and Huy L. Nguyen. 2013. On the Convergence of the Hegselmann-Krause System. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science* (Berkeley, California, USA) *(ITCS '13)*. Association for Computing Machinery, New York, NY, USA, 61–66. https://doi.org/10.1145/2422436.2422446

[14] Stephen P. Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. 2006. Randomized gossip algorithms. *IEEE Trans. Inf. Theory* 52, 6 (2006), 2508–2530. https://doi.org/10.1109/TIT.2006.874516

[15] Leran Cai and Thomas Sauerwald. 2017. Randomized Load Balancing on Networks with Stochastic Inputs. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017 (LIPIcs, Vol. 80)*, Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Warsaw, Poland, 139:1–139:14. https://doi.org/10.4230/LIPIcs.ICALP.2017.139

[16] Bernard Chazelle. 2012. Natural algorithms and influence systems. *Commun. ACM* 55, 12 (2012), 101–110.

[17] Bernard Chazelle. 2015. Diffusive Influence Systems. *SIAM J. Comput.* 44, 5 (2015), 1403–1442. https://doi.org/10.1137/120882640

[18] Colin Cooper, Robert Elsässer, Hirotaka Ono, and Tomasz Radzik. 2012. Coalescing random walks and voting on graphs. In *ACM Symposium on Principles of Distributed Computing, PODC '12, Funchal, Madeira, Portugal, July 16-18, 2012*, Darek Kowalski and Alessandro Panconesi (Eds.). ACM, Madeira, Portugal, 47–56. https://doi.org/10.1145/2332432.2332440

[19] Colin Cooper, Robert Elsässer, and Tomasz Radzik. 2014. The Power of Two Choices in Distributed Voting. In *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 8573)*, Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias (Eds.). Springer, Copenhagen, Denmark, 435–446. https://doi.org/10.1007/978-3-662-43951-7_37

[20] Colin Cooper, Tomasz Radzik, Nicolas Rivera, and Takeharu Shiraga. 2017. Fast Plurality Consensus in Regular Expanders. In *31st International Symposium on Distributed Computing, DISC 2017, October 16-20, 2017 (LIPIcs, Vol. 91)*, Andréa W. Richa (Ed.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Vienna, Austria, 13:1–13:16. https://doi.org/10.4230/LIPIcs.DISC.2017.13

[21] Colin Cooper and Nicolás Rivera. 2018. Threshold behaviour of discordant voting on the complete graph. *J. Discrete Algorithms* 50 (2018), 10–22. https://doi.org/10.1016/j.jda.2018.07.001

[22] George Cybenko. 1989. Dynamic load balancing for distributed memory multi-processors. *Journal of parallel and distributed computing* 7, 2 (1989), 279–301.

[23] M. DeGroot. 1974. Reaching a Consensus. *J. Amer. Statist. Assoc.* 69, 345 (1974), 118–121.

[24] Alejandro D. Domínguez-García and Christoforos N. Hadjicostis. 2011. Distributed strategies for average consensus in directed graphs. In *50th IEEE Conference on Decision and Control and European Control Conference, 11th European Control Conference, CDC/ECC 2011, December 12-15, 2011*. IEEE, Orlando, FL, USA, 2124–2129. https://doi.org/10.1109/CDC.2011.6160462

[25] David Doty, Mahsa Eftekhari, Leszek Gasieniec, Eric E. Severson, Przemyslaw Uznanski, and Grzegorz Stachowiak. 2021. A time and space optimal stable population protocol solving exact majority. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, February 7-10, 2022*. IEEE, Denver, CO, USA, 1044–1055. https://doi.org/10.1109/FOCS52979.2021.00104

[26] Moez Draief and Laurent Massoulié. 2009. *Epidemics and Rumours in Complex Networks*. Cambridge University Press, London, UK. https://doi.org/10.1017/CBO9780511806018

[27] Dimitris Fotakis, Anthimos Vardis Kandiros, Vasilis Kontonis, and Stratis Skoulakis. 2018. Opinion Dynamics with Limited Information. In *Web and Internet Economics - 14th International Conference, WINE 2018, Oxford, UK, December 15-17, 2018, Proceedings (Lecture Notes in Computer Science, Vol. 11316)*, George Christodoulou and Tobias Harks (Eds.). Springer, Oxford, UK, 282–296. https://doi.org/10.1007/978-3-030-04612-5_19

[28] Dimitris Fotakis, Dimitris Palyvos-Giannas, and Stratis Skoulakis. 2016. Opinion Dynamics with Local Interactions. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16)*. AAAI Press, New York, New York, USA, 279–285.

[29] Noah E Friedkin and Eugene C Johnsen. 1990. Social influence and opinions. *Journal of Mathematical Sociology* 15, 3-4 (1990), 193–206.

[30] Tobias Friedrich and Thomas Sauerwald. 2009. Near-perfect load balancing by randomized rounding. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, May 31 - June 2, 2009*, Michael Mitzenmacher (Ed.). ACM, Bethesda, MD, USA, 121–130. https://doi.org/10.1145/1536414.1536433

[31] Javad Ghaderi and R. Srikant. 2014. Opinion dynamics in social networks with stubborn agents: Equilibrium and convergence rate. *Autom.* 50, 12 (2014), 3209–3215. https://doi.org/10.1016/j.automatica.2014.10.034

[32] Mohsen Ghaffari and Merav Parter. 2016. A Polylogarithmic Gossip Algorithm for Plurality Consensus. In *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing, PODC 2016, Chicago, IL, USA, July 25-28, 2016*, George Giakkoupis (Ed.). ACM, Chicago, IL, USA, 117–126. https://doi.org/10.1145/2933057.2933097

[33] Y. Hassin and D. Peleg. 2001. Distributed Probabilistic Polling and Applications to Proportionate Agreement. *Information and Computation* 171, 2 (2001), 248–268. https://doi.org/10.1006/inco.2001.3088

[34] Rainer Hegselmann and Ulrich Krause. 2002. Opinion Dynamics and Bounded Confidence Models, Analysis and Simulation. *Journal of Artificial Societies and Social Simulation* 5, 3 (2002), 1–2. https://ideas.repec.org/a/jas/jasssj/2002-5-2.html

[35] David Kempe, Alin Dobra, and Johannes Gehrke. 2003. Gossip-Based Computation of Aggregate Information. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Proceedings*. IEEE Computer Society, Cambridge, MA, USA, 482–491. https://doi.org/10.1109/SFCS.2003.1238221

[36] Dariusz R. Kowalski and Miguel A. Mosteiro. 2018. Polynomial Counting in Anonymous Dynamic Networks with Applications to Anonymous Dynamic Algebraic Computations. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018 (LIPIcs, Vol. 107)*, Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Prague, Czech Republic, 156:1–156:14. https://doi.org/10.4230/LIPIcs.ICALP.2018.156

[37] Dariusz R. Kowalski and Miguel A. Mosteiro. 2019. Polynomial Anonymous Dynamic Distributed Computing Without a Unique Leader. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019 (LIPIcs, Vol. 132)*, Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Patras, Greece, 147:1–147:15. https://doi.org/10.4230/LIPIcs.ICALP.2019.147

[38] Patrick Lambein-Monette. 2020. *Average consensus in anonymous dynamic networks: An algorithmic.* Ph.D. Dissertation. Polytechnic Institute of Paris, France. https://tel.archives-ouvertes.fr/tel-03168053

[39] Erez Lieberman, Christoph Hauert, and Martin A Nowak. 2005. Evolutionary dynamics on graphs. *Nature* 433, 7023 (2005), 312.

[40] Giuseppe Antonio Di Luna and Roberto Baldoni. 2015. Brief Announcement: Investigating the Cost of Anonymity on Dynamic Networks. In *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015, July 21 - 23, 2015*, Chryssis Georgiou and Paul G. Spirakis (Eds.). ACM, Donostia-San Sebastián, Spain, 339–341. https://doi.org/10.1145/2767386.2767442

[41] Andrea Montanari and Amin Saberi. 2009. Convergence to Equilibrium in Local Interaction Games. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25-27, 2009, Atlanta, Georgia, USA*. IEEE Computer Society, Atlanta, Georgia, USA, 303–312. https://doi.org/10.1109/FOCS.2009.64

[42] Damon Mosk-Aoyama and Devavrat Shah. 2006. Computing separable functions via gossip. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing, PODC 2006, Denver, CO, USA, July 23-26, 2006*, Eric Ruppert and Dahlia Malkhi (Eds.). ACM, Denver, CO, USA, 113–122. https://doi.org/10.1145/1146381.1146401

[43] S. Muthukrishnan, Bhaskar Ghosh, and Martin H. Schultz. 1998. First- and Second-Order Diffusive Methods for Rapid, Coarse, Distributed Load Balancing. *Theory Comput. Syst.* 31, 4 (1998), 331–354. https://doi.org/10.1007/s002240000092

[44] Yuval Rabani, Alistair Sinclair, and Rolf Wanka. 1998. Local Divergence of Markov Chains and the Analysis of Iterative Load Balancing Schemes. In *39th Annual Symposium on Foundations of Computer Science, FOCS '98, November 8-11, 1998, Palo Alto, California, USA*. IEEE Computer Society, Palo Alto, California, USA, 694–705. https://doi.org/10.1109/SFCS.1998.743520

[45] Thomas Sauerwald and He Sun. 2012. Tight Bounds for Randomized Load Balancing on Arbitrary Network Topologies. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, October 20-23, 2012*. IEEE Computer Society, New Brunswick, NJ, USA, 341–350. https://doi.org/10.1109/FOCS.2012.86

[46] Daniel Silvestre, João Pedro Hespanha, and Carlos Silvestre. 2019. Broadcast and Gossip Stochastic Average Consensus Algorithms in Directed Topologies. *IEEE Trans. Control. Netw. Syst.* 6, 2 (2019), 474–486. https://doi.org/10.1109/TCNS.2018.2839341

[47] Raghu Subramanian and Isaac D. Scherson. 1994. An Analysis of Diffusive Load-Balancing. In *Proceedings of the 6th Annual ACM Symposium on Parallel Algorithms and Architectures, SPAA '94, Cape May, New Jersey, USA, June 27-29, 1994*, Lawrence Snyder and Charles E. Leiserson (Eds.). ACM, Cape May, New Jersey, USA, 220–225. https://doi.org/10.1145/181014.181361

[48] J. Tsitsiklis. 1984. *Problems in Decentralized Decision Making and Computation*. Ph.D. Dissertation. MIT.