Discordant voting protocols for cyclically linked agents

András Pongrácz*

Department of Algebra and Number Theory
University of Debrecen
Egyetem suare 1, Debrecen, Hungary 4032

pongracz.andras@science.unideb.hu

Abstract

Voting protocols, such as the push and the pull protocol, model the behavior of people during an election. These processes have been studied in distributed computing in peer-to-peer networks, and to describe how viruses or rumors spread in a community. We determine the asymptotic behavior of the runtime of discordant linear protocols on the cycle graph, and the probability for each consensus to win.

Mathematics Subject Classifications: 91A22, 60J10, 60G50

1 Introduction

Models of voting in finite graphs have been studied intensively for decades, see e.g., [7, 15, 13, 1, 14, 8]. Throughout this paper, a discrete time voting protocol is defined by specifying a graph and a set of nondeterministic rules. Then the process is divided into rounds. In each round, the participants, i.e., vertices of the graph, can affect the vote of their neighbors according to the given rules.

We note that many alternative definitions were investigated in the literature. Continuous time voting processes were studied in [7, 10]. Somewhat surprisingly, the thorough mathematical investigation of the continuous version preceded that of the discrete analogue of the protocols [15, 10]. In [12] the graph evolves together with the opinions of the vertices. This models the behavior of people who in each round try to convince one another and succeed with a given probability. Whenever they fail, they cease to communicate with each other, that is, we delete the edge linking them from the graph. In such a model there are many potential final results, as the graph can disconnect, and in

*This work is supported by the EFOP-3.6.2-16-2017-00015 project, which has been supported by the European Union, co-financed by the European Social Fund. The paper was also supported by the National Research, Development and Innovation Fund of Hungary, financed under the FK 124814 and PD 125160 funding schemes, the János Bolyai Research Scholarship of the Hungarian Academy of Sciences, and by the ÚNKP-18-4 New National Excellence Program of the Ministry of Human Capacities.

fact we may end up with many connected components. For more details, see [8, 2]. The application of these randomized protocols in studying how rumor spreads in a society goes back to decades, and it is still an active area [11, 13, 1]. The same can be said about peer-to-peer networks, see e.g., [18, 14, 3]. In this application, opinion is replaced by a piece of information that each computer has at a given time, and they share the data in a randomized way. Connections of voting processes and coalescing random walks were investigated in [10, 16], and for other recent applications see [17, 5].

However, we consider discrete time voting models where the graph is fixed, and the vote is a binary decision. The two options to choose from are 0 and 1, but we usually refer to vertices with opinion 0 as *blue* vertices, and *red* vertices are the ones with opinion 1. Such a protocol can be synchronous (see [6] for examples), i.e., it is allowed that several vertices of the graph change their opinion in one round; otherwise it is asynchronous. The so-called linear voting model was introduced in [6] as a common generalization of many well-studied voting protocols. Three of the most common special cases of asynchronous linear voting are the

- Oblivious protocol: in each round an edge uv is chosen uniformly at random, and then either u adopts the opinion of v or the other way around, with equal probability.

- Push protocol: in each round a vertex u is chosen uniformly at random, and that vertex forces a randomly chosen neighbor to adopt the opinion of u.

- Pull protocol: in each round a vertex u is chosen uniformly at random, and that vertex is forced by a randomly chosen neighbor v to adopt the opinion of v.

From a practical viewpoint, all linear voting models have a common weakness: it is typical that nothing changes in many steps of the process, as it is possible that every participant keeps his own opinion for the next round. E.g., consider push, pull or oblivious voting on the complete graph K_n; in this particular case, the three protocols coincide. If one opinion is significantly more popular than the other, then with very high probability, both chosen vertices have the more popular opinion. So usually many idle rounds go by before the opinion of some vertex is altered. This example demonstrates the advantage of discordant (oblivious, push, pull) voting protocols, defined in [4]. An edge uv is discordant if u and v have different opinion, and a vertex is discordant if it is in a discordant edge. To define discordant oblivious, push and pull voting, the above three definitions are modified so that whenever a random choice is made, we only allow discordant edges or vertices to be picked (always uniformly at random). Note that in our restricted framework when there are only two opinions, the definition of discordant pull voting simplifies to picking a discordant vertex in each round randomly and switching its opinion.

The goal of every voting scheme that we study now is to reach consensus, that is, a state where all participants have the same opinion. The topic of the present paper is the expected time T to reach consensus with the discordant push, pull and oblivious processes on the n-cycle. It was proven in [4] that all three processes have a quadratic runtime at worst. In particular, push voting is expected to terminate in at most $33n^2$ steps regardless of the initial state, and from some initial state it is indeed expected to take at least $n^2/4 + O(n)$ time to reach a unanimous vote [4, Section 4]. We improve the bounds and obtain the precise asymptotical behavior of the expected runtime of the three discordant protocols on the n-cycle. It is shown that the expected time T for all three asynchronous protocols to reach consensus on the cycle graph with n vertices satisfies $|T - \beta\varrho| = O(n^{3/2})$, where β and ϱ are the number of blue and red vertices in the initial state, respectively. In other words, on the n-cycle $T_{\text{oblivious}}, T_{\text{push}}$ and T_{pull} differ in an $O(n^{3/2})$ term, which is negligible compared to the typically quadratic runtime. The result combined with the lower estimation shows that the worst case is $\beta \sim \rho \sim n/2$, where the expected time is asymptotically $T \sim n^2/4$.

The other vital problem in case of a random protocol is to compute the probability of each possible outcome to win. We show that in case of the cycle graph the probability of each opinion to win with the discordant push, pull or oblivious protocol is asymptotically proportionate to the number of vertices with that opinion in the initial state, provided that the initial state be tame. More precisely, if k denotes the number of runs, i.e., maximal sets of consecutive vertices of the same color, we prove that the blue vertices have winning probability $\beta/n+O(k/n)$. By using some probability theory, it can be shown that there must be a state for arbitrarily large n such that the estimation β/n has error 0.1 or more. However, computer simulations suggest that in highly symmetrical initial states (such as the one with alternating runs of lengths one and two), the estimation β/n is quite accurate, a phenomenon we cannot explain yet.

Although some parts of the proof of the positive results require elaborate combinatorial and probabilistic arguments, the core is an elementary linear algebraic lemma (Lemma 1). This paper is a demonstration of how the iterative application of that elementary lemma can yield asymptotically sharp results to basic questions about evolutionary processes, where the transition matrix is typically large but sparse and easy to describe.

2 Preliminaries

2.1 General tools

Throughout this section, P is an absorbing Markov chain with transient states Tran. We denote by Pen the set of potential *penultimate states* in Tran, that is, the states $t \in \text{Tran}$ such that the probability of moving from t to an absorbing state in one step is positive. As usual, we denote by Q the upper left minor of the canonical form of $P = \begin{pmatrix} Q & R \\ 0 & I \end{pmatrix}$. So Q is the transition matrix restricted to the transient states. Following standard notations, $N = (I-Q)^{-1}$ denotes the fundamental matrix of the Markov chain. In this paper, vectors are column vectors of length $|\,\text{Tran}\,|$, usually denoted by $\underline{u}, \underline{v}, \underline{\varepsilon}$, etc. The coordinates are identified with the transient states, so precisely speaking, these are vectors in $\mathbb{R}^{\text{Tran}}$. We denote by $\underline{1}$ the column vector of length $|\,\text{Tran}\,|$ all of whose entries equal to 1. The entry corresponding to the coordinate t in the vector $\underline{u}$ is denoted by $u[t]$. It is well-known that if we sum up the entries $u[t]$ while randomly walking on the coordinates starting from $t_0 \in \text{Tran}$, then the expected value of this sum before the walk is absorbed is $(N\underline{u})[t_0]$. In particular, the expected times to absorption from each transient state as initial state are the coordinates of the vector $N\underline{1}$.

The following lemma is the basic observation of the elementary method we use to improve the upper estimations for the expected time to absorption presented in [4]. We can think about $x[t]$ as a "guesstimate" of the expected value of the sum of the entries of $\underline{u}$ during a random walk with initial state t before reaching an absorbing state. In particular, if $\underline{u} = \underline{1}$, then $\underline{x}$ is the guesstimate vector for the time to absorption starting from each transient state.

Lemma 1. *Let $\underline{u}, \underline{x}, \underline{\varepsilon} \in \mathbb{R}^{\mathrm{Tran}}$ be vectors such that $Q\underline{x} = \underline{x} - \underline{u} + \underline{\varepsilon}$. Then $N\underline{u} = \underline{x} + N\underline{\varepsilon}$. In particular, if $Q\underline{x} \leqslant \underline{x} - \underline{u}$, then $N\underline{u} \leqslant \underline{x}$ (coordinate-wise).*

Proof. By rearranging the equation we obtain $\underline{u} = (I - Q)\underline{x} + \underline{\varepsilon}$. Multiplying both sides by $N = (I - Q)^{-1}$ yields $N\underline{u} = \underline{x} + N\underline{\varepsilon}$.

Moreover, $N = (I-Q)^{-1} = I + Q + Q^2 + \cdots$ is a non-negative matrix. Hence, if $\underline{\varepsilon} \leqslant \underline{0}$ coordinate-wise, then $N\underline{\varepsilon} \leqslant N\underline{0} = \underline{0}$, thus $N\underline{u} \leqslant \underline{x}$. $\square$

As we mentioned earlier, the vectors $N\underline{u}$ and $N\underline{\varepsilon}$ are the expected value vectors of the sum of the entries of $\underline{u}$ and $\underline{\varepsilon}$ during a random walk (on the coordinates) starting from each transient state. The above elementary lemma is particularly useful when the transition matrix is large but sparse, and the fundamental matrix cannot be computed or represented in a transparent way. This is often the case with evolutionary processes. Note that $Q\underline{x}$ is easy to compute if the matrix is sparse. Furthermore, because of the probabilistic interpretation of $N\underline{\varepsilon}$ and the possibility of applying Lemma 1 iteratively, it is possible to estimate this vector without computing N, as we see later. By successive application of this method, the error can shrink to such a small vector that it is very easy to estimate it, providing us with an efficient estimation of the expected value vector. We spell out an immediate application.

Lemma 2. *Let $\underline{u} \in \mathbb{R}^{\mathrm{Tran}}$ be such that $u[t] = 0$ for all $t \in \mathrm{Tran} \setminus \mathrm{Pen}$. Define $\underline{p} \in \mathbb{R}^{\mathrm{Tran}}$ where $p[t]$ is the probability of immediate absorption in state t. Let $M := \max_{t \in \mathrm{Pen}} u[t]/p[t]$. Then the expected sum of the entries of $\underline{u}$ during a random walk from any initial state is at most M.*

Proof. Apply Lemma 1 with the guesstimate vector $\underline{x} = M \cdot \underline{1}$. Note that $\underline{u} \leqslant M \cdot \underline{p}$, thus $\underline{x} - \underline{u} \geqslant M \cdot (\underline{1} - \underline{p}) = M \cdot Q\underline{1} = Q\underline{x}$. Hence, $N\underline{u} \leqslant \underline{x}$ coordinate-wise by Lemma 1. $\square$

This observation is very advantageous when we are able to cut a process to several phases, and we want to estimate the expected sum of an expression between two phase transitions. In our case, the phases are those parts of the process where the number of *runs*, i.e., maximal sets of consecutive vertices with the same opinion in the cycle, is constant. Note that the number of runs cannot increase during the process, and it decreases by two whenever the opinion of a singleton vertex is switched. The only exception is when we reach consensus in the last step: in that case, the number of runs drops down from two to one.

2.2 Further terminology

We now turn to the problems under consideration, defined in the introduction. Note that the proof is presented for discordant push voting on the n-cycle: the case of pull voting can be done in a similar fashion, and the case of oblivious voting is trivial. Clearly, the voting process is an absorbing Markov chain with 2^n states, whose absorbing states are exactly those two where all the vertices agree.

As in the introduction, the number of blue and red vertices are denoted by β and ϱ, respectively. A vertex is a *singleton* if its color differs from both its neighbors' color. The number of singleton blue and red vertices are s_β and s_ϱ, respectively. The number of non-singleton blue vertices with (exactly) one red neighbor is m_β; the number m_ϱ is defined analogously for red vertices.

Note that the number of runs is even in every state, except for the two absorbing states where the whole cycle is one run. Furthermore, the number of red runs equals to the number of blue runs in the transient states, as red and blue runs alternate in the cycle. A maximal set of consecutive singleton vertices is called an *arc*, and the number of arcs is denoted by ℓ.

3 Expected time to absorption on the cycle

It turns out to be advantageous in the calculation to cut the process into two parts. We fix a number $K = 8\sqrt{5}\sqrt{n}$, and then the first part of the process consists of the steps before we first reach a state with K runs.

3.1 The first part: down to K runs

In this subsection, we show an estimation of the expected length of the first part. The following bound can be extracted from [4, Section 4]. In that paper, a quadratic upper estimation was given to the runtime of the discordant push protocol using some results about stopped martingales. They obtained that it takes at most $33n^2$ steps to reach consensus from any initial state, that is, to reach a state with one run. However, by carefully modifying their calculations, a more general result can be shown.

Proposition 3. *The expected time to reach a state with k runs is at most $80n^2/k$ from any initial state. In particular, putting $k = K = 8\sqrt{5}\sqrt{n}$, the first part is expected to terminate in at most $2\sqrt{5} \cdot n^{3/2}$ steps.*

Proof. In [4, Lemma 8] and the argument before that, it was shown that the expected time to reach a state with $k = 2r_1$ runs from one with $2r_0$ runs is at most T^*, where T^*

is the optimal solution of the following linear program:

$$T^* = \max 10\sqrt{2}n^{3/2} \sum_{r=r_1}^{r_0} \frac{x_r}{r^{3/2}}$$

$$\text{such that} \quad \sum_{j=r_1}^{r} x_j \leqslant \sqrt{2rn} \quad \text{for all} \quad r_1 \leqslant r \leqslant r_0$$
$$\text{and} \quad x_r \geqslant 0 \quad \text{for all} \quad r_1 \leqslant r \leqslant r_0$$

Moreover, it can be shown that such a linear program attains its optimal solution at $x_{r_1} = \sqrt{2r_1n}$ and $x_r = \sqrt{2rn} - \sqrt{2(r-1)n}$ for all $r_1 + 1 \leqslant j \leqslant r_0$. Hence, by using the standard estimations $\sqrt{2rn} - \sqrt{2(r-1)n} \leqslant \sqrt{\frac{2n}{r}}$ and $\frac{1}{r^2} \leqslant \frac{1}{r(r-1)} = \frac{1}{r-1} - \frac{1}{r}$ we obtain

$$T^* \leqslant 10\sqrt{2}n^{3/2}\left(\frac{\sqrt{2r_1n}}{r_1^{3/2}} + \sum_{r=r_1+1}^{r_0} \frac{\sqrt{2rn} - \sqrt{2(r-1)n}}{r^{3/2}}\right) \leqslant$$
$$10\sqrt{2}n^{3/2}\left(\frac{\sqrt{2n}}{r_1} + \sum_{r=r_1+1}^{r_0} \frac{\sqrt{2n}}{r^2}\right) = 20n^2\left(\frac{1}{r_1} + \sum_{r=r_1+1}^{r_0} \frac{1}{r^2}\right) \leqslant$$
$$20n^2\left(\frac{1}{r_1} + \sum_{r=r_1+1}^{r_0}\left(\frac{1}{r-1} - \frac{1}{r}\right)\right) = 20n^2\left(\frac{1}{r_1} + \frac{1}{r_1} - \frac{1}{r_0}\right) \leqslant \frac{40n^2}{r_1}$$

Substituting $r_1 = \frac{k}{2}$ finishes the proof. $\square$

3.2 The second part: from K runs to consensus

We begin with a technical lemma.

Lemma 4. *The expected value of the sum of* $|\frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho}|$ *during a random walk until the number of runs decreases is at most* $1/2$. *In particular, the expected sum of the above expression during the second part of the voting process is at most* $2\sqrt{5}\sqrt{n}$.

Proof. We use Lemma 2. In order to do that, the Markov chain is restricted to those states that have k runs, and extended by an absorbing state where we move exactly when in the original Markov chain the number of runs decreases. The penultimate states of this chain are exactly those states with k runs in our problem where there is a singleton vertex. Of course, the number $p[t]$ (the probability of immediate absorption from t) is the probability that we lose runs, which is the probability that in the original Markov chain a singleton is pushed. Our next goal is to calculate the probability of this event.

We call the states with alternating red and blue vertices *special* states. Such states exist iff n is even, and then there are two of them. Now assume that the state is not special. Let $a_1, \ldots, a_h$ be an arc, surrounded by the non-singleton vertices b (a neighbor of a_1) and c (a neighbor of a_h). As we are not in the special states, the arc is not the full set of vertices, and b and c are indeed not singletons. We show that the probability that a singleton be pushed in this arc is $\frac{h+1}{d}$, where $d = s_\beta + s_\varrho + m_\beta + m_\varrho$ is the number of discordant vertices. If $h = 1$, then the vertex a_1 is indeed pushed with probability $\frac{2}{d}$: this happens exactly when b or c is chosen out of the d discordant vertices for pushing their opinion. If $h \geqslant 2$, then a vertex in the arc is pushed iff

- a vertex is chosen out of $b, a_1, \ldots, a_h, c$ for pushing, and
- if that vertex is a_1 or a_h, then the singleton neighbor is chosen.

Hence, the desired probability is $\frac{h+2}{d} - \frac{2}{d} \cdot \frac{1}{2} = \frac{h+1}{d}$. Adding the expression $\frac{h+1}{d}$ for all arcs, we obtain that the probability that a singleton vertex be pushed in the state t is $p[t] = \frac{s_\beta + s_\varrho + \ell}{s_\beta + s_\varrho + m_\beta + m_\varrho}$, if t is not a special state.

Let the vector $\underline{u}$ have entries $|\frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + s_\varrho + m_\beta + m_\varrho}|$ for each transient state. Note that this is 0 for non-penultimate transient states, since red and blue runs alternate, so $s_\beta = s_\varrho = 0$ and $m_\beta = m_\varrho$. It is also 0 for the two special states.

So the expression $\frac{u[t]}{p[t]}$ (cf. Lemma 2) equals to $|\frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + s_\varrho + \ell}|$ for all penultimate states. By Lemma 2 it suffices to show that $\frac{1}{2}$ is an upper bound for this expression. Observe that the expression does not decrease if we double two singleton vertices of opposite color. That is, we replace the two vertices by inserting two edges at the same positions in the cycle, obtaining a new cycle of length $n+2$, and coloring the endpoints of the edge replacing the red and the blue vertex red and blue, respectively. Indeed, the numerator is not modified by this operation, and the denominator cannot increase, as $s_\beta + s_\varrho$ decreases by 2, and ℓ increases by at most 2. After a finite number of applications of this operation, we reach a state where all singleton vertices have the same color, say blue. In particular, there are no consecutive singleton vertices in the cycle. Thus $s_\beta = \ell$, $s_\varrho = 0$ and $m_\varrho = m_\beta + 2s_\beta$, so the expression simplifies to $|\frac{-s_\beta}{2s_\beta}| = \frac{1}{2}$. If all singletons disappear after a finite number of applications of the above operation, then the numerator of the expression is 0, thus it has been 0 when we started eliminating singletons, as well.

The second assertion of the lemma follows easily, as the number of positive even numbers below $8\sqrt{5}\sqrt{n}$ is at most $4\sqrt{5}\sqrt{n}$. □

As we suggested earlier, it seems impossible to compute the fundamental matrix of our Markov chain. However, the upper-left minor Q of the transition matrix is sparse, so Lemmas 1 and 2 can be applied. The way we phrased the result in the introduction provides the right heuristics for the guesstimate vector. The expected runtime of the oblivious protocol is clearly $\beta\varrho$: it is simply the runtime of a drunkard walk with parameter $n = \beta + \varrho$ and initial state β (see [4] for details). Computer simulations (in SAGE) suggested that the runtime of the three discordant protocols should be close to each other. The intuitive reason is that the transition matrix of the three protocols on the cycle graph coincide in almost all entries. Of course, such an observation can lead to very badly wrong conjectures in general, as the computation of the fundamental matrix involves the calculation of an inverse matrix, which is very sensitive to even small alterations of a few entries of the matrix. Hence, in order to turn this intuition into a precise proof, we use Lemma 1 with guesstimate vector $\underline{x}$ whose entries are $\beta\varrho$ for each transient state.

Theorem 5. *Given any initial state on an n-cycle with β blue and ϱ red vertices. Let T be the expected number of steps for the discordant push voting to reach consensus. Then $|T-\beta\varrho| \leqslant 4\sqrt{5}n^{3/2}$. In particular, the worst expected runtime is asymptotically $n^2/4$, obtained when $\beta \sim n/2$.*

Proof. Let us assume that the first half of the process has terminated. By Proporsition 3 this is expected to take $T_1 \leqslant 2\sqrt{5}n^{3/2}$ steps.

Let $\underline{x}$ be the column vector of length 2^n-2 with coordinates $\beta\varrho$ for each transient state. Then the probability of the number of blue vertices to increase by 1, i.e., a blue vertex is pushing, is $\frac{s_\beta+m_\beta}{s_\beta+m_\beta+s_\varrho+m_\varrho}$. Similarly, the probability of the number of red vertices to increase by 1 is $\frac{s_\varrho+m_\varrho}{s_\beta+m_\beta+s_\varrho+m_\varrho}$. If we multiply the value of the vector $\underline{x}$ with the corresponding transition probabilities, and add them up, i.e., we calculate $Q\underline{x}$ (cf. Lemma 1), we obtain:

$$\frac{s_\beta+m_\beta}{s_\beta+m_\beta+s_\varrho+m_\varrho}(\beta+1)(\varrho-1)+\frac{s_\varrho+m_\varrho}{s_\beta+m_\beta+s_\varrho+m_\varrho}(\beta-1)(\varrho+1)=$$

$$\beta\varrho-1+\frac{(\varrho-\beta)(s_\beta+m_\beta-s_\varrho-m_\varrho)}{s_\beta+m_\beta+s_\varrho+m_\varrho}$$

for all non-penultimate transient states. Using the notations of Lemma 1 with $\underline{u}=\underline{1}$, the entry of the error vector $\underline{\varepsilon}=Q\underline{x}-\underline{x}+\underline{1}$ at the given state is $\frac{(\varrho-\beta)(s_\beta+m_\beta-s_\varrho-m_\varrho)}{s_\beta+m_\beta+s_\varrho+m_\varrho}$, whose absolute value is at most $n|\frac{s_\beta+m_\beta-s_\varrho-m_\varrho}{s_\beta+m_\beta+s_\varrho+m_\varrho}|$. To obtain the error for penultimate transient states, we calculate it when there is exactly one red vertex, i.e., $s_\varrho=1, m_\varrho=0, s_\beta=0, m_\beta=2, \rho=1, \beta=n-1$. (The situation when there is exactly one blue vertex is analogous.) In that state $Q\underline{x}=\frac{s_\varrho+m_\varrho}{s_\beta+m_\beta+s_\varrho+m_\varrho}(\beta-1)(\varrho+1)=\frac{2n-4}{3}$, thus $\underline{\varepsilon}=Q\underline{x}-\underline{x}+1$ has entry $\frac{2n-4}{3}-(n-1)+1=-\frac{n-2}{3}$. The absolute value of this number is at most $\frac{n}{3}=n|\frac{s_\beta+m_\beta-s_\varrho-m_\varrho}{s_\beta+m_\beta+s_\varrho+m_\varrho}|$ again. Hence, $n|\frac{s_\beta+m_\beta-s_\varrho-m_\varrho}{s_\beta+m_\beta+s_\varrho+m_\varrho}|$ estimates the absolute value of the error at all transient states from above, and the expected sum of this expression during a random walk is at most $2\sqrt{5}n^{3/2}$ by Lemma 4.

If T_2 denotes the expected runtime of the second part of the process, then $|T_2-\beta\varrho| \leqslant 2\sqrt{5}n^{3/2}$ by Lemma 1. As $T=T_1+T_2$, $|T-\beta\varrho|=|T_1+T_2-\beta\varrho| \leqslant |T_1|+|T_2-\beta\varrho|$ yields the desired estimation. □

Remark 6. In a similar fashion, it can be shown that the expected time for the discordant pull voting to reach consensus on the cycle is also $\beta\rho+O(n^{3/2})$. It is clear that switching the cycle to a path makes very little difference in the calculation, and estimations of the same order of magnitude are obtained in case of the discordant push, pull and oblivious protocols on paths, too.

4 Winning probabilities on the cycle

It is enough to estimate the winning probability p of the color blue, the other color then wins with probability $1-p$. Again, we know from standard theory that the matrix NR

consists of the probabilities of reaching from transient state i the absorbing state j in the process. So we are only interested in the first column of this $(2^n - 2) \times 2$ matrix. Lemma 1 can be applied, as the problem is to estimate the vector $N\underline{u}$ where $\underline{u}$ is the first column of R.

Theorem 7. *Given any initial state on an n-cycle with β blue and ϱ red vertices, and with k runs. Let p be the probability that the blue consensus is reached with the discordant push protocol. Then $|p - \beta/n| \leqslant k/4n$.*

Proof. Let $\underline{x}$ be the column vector of length $2^n - 2$ with coordinates $\frac{\beta}{n}$ for each transient state. If we multiply the value of the vector $\underline{x}$ with the corresponding transition probabilities (cf. the proof of Theorem 5), and add them up, i.e., we calculate $Q\underline{x}$, we obtain:

$$\frac{s_\beta + m_\beta}{s_\beta + m_\beta + s_\varrho + m_\varrho} \cdot \frac{\beta + 1}{n} + \frac{s_\varrho + m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho} \cdot \frac{\beta - 1}{n} =$$
$$\frac{\beta}{n} + \frac{1}{n} \cdot \frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho}$$

for all non-penultimate transient states. We use the notations of Lemma 1 with $\underline{u}$ being the all 0 vector, except for the entries corresponding to the states with exactly one red vertex which are all $\frac{2}{3}$. In particular, for non-penultimate states the error, i.e., the corresponding entry of $\underline{\varepsilon} = Q\underline{x} - \underline{x} + \underline{u}$ is $\frac{1}{n} \cdot \frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho}$.

For penultimate states with exactly one red vertex, the entry of the vector $Q\underline{x}$ is $\frac{s_\varrho + m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho} \cdot \frac{\beta - 1}{n} = \frac{n-2}{3n}$. Thus the error is $\frac{n-2}{3n} - \frac{n-1}{n} + \frac{2}{3} = \frac{1}{3n}$ which is precisely $\frac{1}{n} \cdot \frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho}$.

Finally, for penultimate states with exactly one blue vertex, the entry of the vector $Q\underline{x}$ is $\frac{s_\beta + m_\beta}{s_\beta + m_\beta + s_\varrho + m_\varrho} \cdot \frac{\beta + 1}{n} = \frac{2}{3n}$. Thus the error is $\frac{2}{3n} - \frac{1}{n} + 0 = -\frac{1}{3n} = \frac{1}{n} \cdot \frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho}$.

Hence, the vector with entries $\frac{1}{n} \cdot \frac{s_\beta + m_\beta - s_\varrho - m_\varrho}{s_\beta + m_\beta + s_\varrho + m_\varrho}$ is exactly the error vector $\underline{\varepsilon}$. As the number of runs decreases $\frac{k}{2}$ times before a consensus is reached, the assertion follows by Lemma 1 and Lemma 4. □

Theorem 7 shows that if $k = o(n)$, then the probability to reach the blue consensus with the linear push voting and the discordant push voting protocols have the same asymptotics as $n \to \infty$. However, the same result cannot hold when k is large. To show a reasonably large gap between the probabilities arising from the linear and discordant push protocols, we recall a technical tool.

Theorem 8 (Chernoff bound)**.** *Let $X_1, \ldots, X_n$ be random variables such that $a \leqslant X_i \leqslant b$ for some $a, b \in \mathbb{R}$. Let $Y = \sum_{i=1}^{n} X_i$, $\mu = E(Y)$, and let $\delta > 0$. Then $P(Y \leqslant (1-\delta)\mu) \leqslant \exp(-\frac{\delta^2 \mu^2}{n(b-a)^2})$.*

Lemma 9. *Let n be divisible by 6. If we run the discordant push protocol on the n-cycle from the state where all red runs have length 1 and all blue runs have length 2, then the* *ratio of blue and red vertices after $i \leqslant n/6$ steps is at least $\frac{2n/3-i}{n/3+i}$. Moreover, we have $\sum_{i=1}^{n/6} \frac{2n/3-i}{n/3+i} \geqslant 0.238 \cdot n$ if $n \geqslant 2000$.*

Proof. In any asynchronous protocol (i.e., when in one step only one vertex alters its opinion), the following analysis applies.

Case 1: If the number of runs does not change in a step. If color c is spreading, then the number of discordant vertices of color c changes by 0 or $+1$, and the number of discordant vertices of the other color changes by 0 or -1. The worst such case in terms of the discordant blue-red ratio is thus if the number of discordant blue vertices decreases by 1 and the number of discordant red vertices increases by 1. Note that this is in fact possible if a singleton red vertex pushes an endpoint of a blue run of length 2.

Case 2: If the number of runs decreases in a step. If color c is spreading, then the number of discordant vertices of color c changes by $-2, -1$ or 0 , and the number of discordant vertices of the other color changes by -1. The worst such case in terms of the discordant blue-red ratio is thus if the number of discordant blue vertices decreases by 2 and the the number of discordant red vertices decreases by 1. Note that this is in fact possible if a singleton red vertex is pushed.

Let a be the number of steps as in Case 1, and b as in Case 2 out of the first i steps. Then the discordant blue-red ratio after i steps is at least $\frac{2n/3-a-2b}{n/3+a-b} = \frac{2n/3-i-b}{n/3+i-2b} = \frac{1}{2} + \frac{n/2-3i/2}{n/3+i-2b}$ which as a function of b with domain $0, 1, \ldots, \frac{n}{6}$ is strictly monotone increasing. Hence, the lowest possible ratio is attained after i steps if $a = i, b = 0$, that is, the worst case described in Case 1 occurs i times, and then the ratio is $\frac{2n/3-i}{n/3+i}$.

To estimate the sum, we denote by $H(k)$ the sum of the harmonic series $\sum_{i=1}^{k} \frac{1}{i}$. We use the bounds $\log k + \gamma \leqslant H(k) \leqslant \log k + \gamma + \frac{1}{2k}$ if $k \geqslant 100$, which follows easily from (9.88) in [9]. Thus

$$\sum_{i=1}^{n/6} \frac{2n/3 - i}{n/3 + i} = \sum_{i=1}^{n/6} \frac{-(n/3+i) + n}{n/3 + i} = -\frac{n}{6} + n \cdot \sum_{i=1}^{n/6} \frac{1}{n/3 + i} =$$

$$= -\frac{n}{6} + n \cdot (H(\frac{n}{2}) - H(\frac{n}{3})) \geqslant -\frac{n}{6} + n \cdot (\log 3 - \log 2) - \frac{3}{2} \geqslant 0.238n$$

if $n \geqslant 2000$. □

Theorem 10. *For any $n \geqslant 20000$ there exists a state on the n-cycle with β blue vertices and probability p of reaching the blue consensus with the discordant push protocol such that $|p - \beta/n| > 0.1$.*

Proof. The proof is indirect; so assume that $|p - \frac{\beta}{n}| \leqslant 0.1$ for all states for some fixed $n \geqslant 20000$. We assume that n is divisible by 6, and show that $|p - \frac{\beta}{n}| \leqslant 0.1005$ cannot hold for all states. Consider the initial state with all red runs of length 1 and all blue runs of length 2. Hence, by the indirect assumption the probability of reaching the blue consensus from this initial state is between $\frac{2}{3} - 0.1005$ and $\frac{2}{3} + 0.1005$.

Now we let the process run for $\frac{n}{6}$ steps. By Lemma 9 the probability p_i that a blue vertex is pushing in the i-th step is at least $\frac{2n/3-i}{n/3+i}$. Let X_i be the increment of blue vertices in the i-th step for $i \leqslant \frac{n}{6}$, and let $Y = \sum_{i=1}^{n/6} X_i$. Then $\mu = E(Y) \geqslant 0.238n$ by Lemma 9. Put $\delta = 0.138$, $a = -1, b = 1$. Then $(1-\delta)\mu \geqslant 0.2052n$, thus Theorem 8 yields $P(Y \leqslant 0.2052n) \leqslant \exp(-\frac{\delta^2\mu^2}{n(b-a)^2}) \leqslant 0.0045$ if $n \geqslant 20000$.

Hence, by the law of total probability and the indirect assumption (by estimating the probability with 0 whenever $Y \leqslant 0.2052n$) we obtain a contradiction as follows: $\frac{2}{3} + 0.1005 \geqslant 0.9955 \cdot (\frac{2}{3} + 0.2052 - 0.1005) \geqslant \frac{2}{3} + 0.1012$.

□

Due to Proposition 3 and Theorem 7 it is however possible to estimate the desired probability up to an error term $O(1/\sqrt{n})$ by running the process for $O(n^{3/2})$ steps. This is a good trade-off as the expected time to reach consensus from the worst initial case is $n^2/4 + O(n^{3/2})$, and it is quadratic in general (cf. Theorem 5). In fact, using this observation, it is possible to write a relatively fast program that runs the experiment on the cycle with 5000 vertices 5000 times from the initial state described in the proof of Theorem 10. The result suggests that in that very symmetrical state, the estimate 2/3 for the probability that the blue consensus be reached is highly accurate. It would be interesting to give a precise proof to this phenomenon (provided it is true), and to explain why the formula works in that particular situation. The proof of Theorem 10 might suggest that such a result is somewhat counter-intuitive. If this empirical result is correct, then there must be a more complicated formula that takes into consideration the position of blue vertices around the cycle as well as their number, and coincidentally this formula should assign a value close to 2/3 to the above mentioned initial state.

5 Further results and future work

It is also possible to obtain asymptotically sharp estimates for the corresponding problems in the star graph with n vertices. This is a typical network, when one server is connected to several clients. It was already pointed out in [4] that, quite counter-intuitively, the discordant pull protocol is faster than the discordant push protocol on such graphs if n is large enough. The author of the present paper together with coauthors were able to refine this result, and obtain asymptotically sharp estimations for both expected runtimes.

References

[1] Acan, H., Collevecchio, A., Mehrabian, A., and Wormald, N. On the push&pull protocol for rumour spreading. In *Proceedings of the ACM Symposium on Principles of Distributed Computing* (2015), pp. 405–412.

[2] Basu, R., and Sly, A. Evolving voter model on dense random graphs. *Annals of Applied Probability 27*, 2 (2017), 1235–1288.

[3] Carra, D., Cigno, R. L., and Russo, A. On some fundamental properties of p2p push/pull protocols. In *Proceedings of the Second International Conference on Communications and Electronics* (2008), p. 7 pp.

[4] Cooper, C., Dyer, M., Frieze, A., and Rivera, N. Discordant voting processes on finite graphs. In *43rd International Colloquium on Automata, Languages, and Programming* (ICALP 2016), pp. 2033–2045.

[5] Cooper, C., Elsasser, R., Ono, H., and Radzik, T. Coalescing random walks and voting on connected graphs. *SIAM Journal on Discrete Mathematics 27*, 4 (2013), 1748–1758.

[6] Cooper, C., and Rivera, N. The linear voting model. In *43rd International Colloquium on Automata, Languages, and Programming* (ICALP 2016), pp. 2021–2032.

[7] Donnelly, P., and Welsh, D. Finite particle systems and infection models. *Mathematical Proceedings of the Cambridge Philosophical Society 94*, 1 (1983), 167–182.

[8] Durrett, R., Gleeson, J. P., Lloyd, A. L., Mucha, P. J., Shi, F., and Sivakoff, D. Graph fission in an evolving voter model. *Proceedings of the National Academy of Sciences 109*, 10 (2012), 3682–3687.

[9] Graham, R. L., Knuth, D. E., and Patashnik, O. *Concrete mathematics: a foundation for computer science*, 2 ed. Addison-Wesley, 1994.

[10] Hassin, Y., and Peleg, D. Distributed probabilistic polling and applications to proportionate agreement. *Information and Computation 171*, 2 (2001), 248–268.

[11] Hedetniemi, S. M., Hedetniemi, S. T., and Liestman, A. L. A survey of gossiping and broadcasting in communication networks. *Networks 18*, 4 (1988), 319–349.

[12] Holme, P., and Newman, M. E. J. Nonequilibrium phase transition in the coevolution of networks and opinions. *Physical Review E 74*, 5 (2006), 5 pp.

[13] Karp, R., Schindelhauer, C., Shenke, S., and Vocking, B. Randomized rumor spreading. In *FOCS '00 Proceedings of the 41st Annual Symposium on Foundations of Computer Science* (2000), pp. 565–574.

[14] Locher, T., Meier, R., Schmid, S., and Wattenhofer, R. Push-to-pull peer-to-peer live streaming. In *Proceedings of the International Symposium on Distributed Computing* (2007), pp. 388–402.

[15] Nakata, T., Imahayashi, H., and Yamashita, M. *Probabilistic local majority voting for the agreement problem on finite graphs*. Springer, 1999.

[16] Oliveira, R. On the coalescence time of reversible random walks. *Transactions of the American Mathematical Society 364*, 4 (2012), 2109–2128.

[17] Oliveira, R. Mean field conditions for coalescing random walks. *The Annals of Probability 41*, 5 (2013), 3420–3461.

[18] Tran, D. A., Hua, K. A., and Do, T. T. A peer-to-peer architecture for media streaming. *IEEE Journal on Selected Areas in Communications 22*, 1 (2004), 121–133.