Resilience of 3-Majority Dynamics to Non-Uniform Schedulers

Uri Meir ✉
Blavatnik School of Computer Science, Tel Aviv University, Israel

Rotem Oshman ✉
Blavatnik School of Computer Science, Tel Aviv University, Israel

Ofer Shayevitz ✉
School of Electrical Engineering, Tel Aviv University, Israel

Yuval Volkov ✉
School of Electrical Engineering, Tel Aviv University, Israel

Abstract

In recent years there has been great interest in networks of passive, computationally-weak nodes, whose interactions are controlled by the outside environment; examples include population protocols, chemical reactions networks (CRNs), DNA computing, and more. Such networks are usually studied under one of two extreme regimes: the schedule of interactions is either assumed to be adversarial, or it is assumed to be chosen uniformly at random. In this paper we study an intermediate regime, where the interaction at each step is chosen from some not-necessarily-uniform distribution: we introduce the definition of a (p, ε)-scheduler, where the distribution that the scheduler chooses at every round can be arbitrary, but it must have ℓ_p-distance at most ε from the uniform distribution. We ask how far from uniform we can get before the dynamics of the model break down.

For simplicity, we focus on the 3-majority dynamics, a type of chemical reaction network where the nodes of the network interact in triplets. Each node initially has an opinion of either X or Y, and when a triplet of nodes interact, all three nodes change their opinion to the majority of their three opinions. It is known that under a uniformly random scheduler, if we have an initial gap of $\Omega(\sqrt{n \log n})$ in favor of one value, then w.h.p. all nodes converge to the majority value within $O(n \log n)$ steps.

For the 3-majority dynamics, we prove that among all non-uniform schedulers with a given ℓ_1- or ℓ_∞-distance to the uniform scheduler, the worst case is a scheduler that creates a *partition* in the network, disconnecting some nodes from the rest: under any (p, ε)-close scheduler, if the scheduler's distance from uniform only suffices to disconnect a set of size at most S nodes and we start from a configuration with a gap of $\Omega(S + \sqrt{n \log n})$ in favor of one value, then we are guaranteed that all but $O(S)$ nodes will convert to the majority value. We also show that creating a partition is not necessary to cause the system to converge to the wrong value, or to fail to converge at all. We believe that our work can serve as a first step towards understanding the resilience of chemical reaction networks and population protocols under non-uniform schedulers.

2012 ACM Subject Classification Computing methodologies → Distributed algorithms; Applied computing → Biological networks

Keywords and phrases chemical reaction networks, population protocols, randomized scheduler

Digital Object Identifier 10.4230/LIPIcs.ITCS.2023.86

Funding *Rotem Oshman*: Research funded by the Israel Science Foundation, Grant No. 2801/20, and also supported by Len Blavatnik and the Blavatnik Family foundation.
Ofer Shayevitz: Research funded by the Israel Science Foundation, Grant No. 1766/22 and 1495/18.

14th Innovations in Theoretical Computer Science Conference (ITCS 2023).
Editor: Yael Tauman Kalai; Article No. 86; pp. 86:1–86:19
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In recent years there has been great interest in various models of passively mobile computation networks, such as biological and biologically-inspired computing, passively-mobile sensor networks, chemical reaction networks, and more. These models typically feature very weak computation nodes, often modeled as finite-state machines, whose interactions with one another are controlled by the environment. The goal is to study the dynamics of the system, or to characterize its computation power. One of the most widely-studied problems in this context is *approximate majority*: if each node starts out with some value or opinion, and there is a noticeable gap in favor of one value, is it guaranteed that through their interactions with each other, the nodes that hold the majority opinion can convert all the other nodes to their opinion? Majority comes up in many contexts, ranging from implementable programming language using DNA nanotechnology [13, 27] to a useful building block in distributed protocols for passively-mobile sensor networks [8].

In this paper we focus on one specific model, 3-majority dynamics [14]: the nodes in the population are denoted $\{1, \ldots, n\}$, and each node has an initial value of either X or Y. The nodes interact in triplets: when a triplet $\{i, j, k\}$ is scheduled, all three nodes i, j, k change their value to the *majority value* of the triplet – e.g., if i, j have value X and k has value Y, then following the interaction $\{i, j, k\}$, all three nodes will have value X. This is in some sense the minimal model under which agreement is possible: e.g., with pairwise interactions, it is known that two possible states for the nodes do not suffice [24].

The environment that controls the interactions of the nodes is modeled as a *scheduler*. Typically, two extreme types of schedulers are considered: *worst-case schedulers*, where the next interaction is chosen adversarially at each step,[1] and *uniformly random schedulers*, where the next interaction is chosen uniformly at random from the set of possible interactions. There is a wealth of results characterizing the parameters under which approximate (or exact) majority can be computed in each setting, as well as the time and memory required to do so [6, 7, 24, 17, 21, 4, 22, 23, 1, 12, 2, 14, 5, 11, 10, 9, 16]. The simplicity of the uniformly random scheduler is appealing from the theoretical point of view, and allows us to tightly characterize the behavior of various system dynamics; however, assuming a perfectly uniform scheduler is perhaps too idealistic, and a-priori, it is not clear whether results proven under this assumption are robust even to small changes in the scheduler's distribution.

In this paper we consider an intermediate regime between worst-case and uniformly random schedulers: we allow the scheduler to choose at each step some distribution μ over the set of possible interactions, and the next interaction is then chosen according to μ. This interpolates between the adversarial regime, which can be modeled by having the scheduler choose a singleton distribution μ, and the uniformly random regime, which can be modeled by always taking $\mu = U$, the uniform distribution. It also includes the natural class of schedulers where the next interaction is always chosen from the same distribution η, but η is not necessarily uniform. Our central question in this paper is: how resilient are approximate majority protocols to non-uniform schedulers? How far from the uniform scheduler is "too far"?

To study this question formally, let us say that a scheduler S is (p, ε)*-close to uniform* if at every step, the distribution μ chosen by S satisfies $||\mu - U||_p \leq \varepsilon$, If μ never changes we call the scheduler *fixed*. here, $||\mu - \eta||_p$ denotes the ℓ_p-distance between distributions, defined by

[1] Usually under some *fairness assumption*, e.g., that every possible interaction is scheduled infinitely often. [8]

$$||\mu - \eta||_p = \left(\sum_{\omega} |\mu(\omega) - \eta(\omega)|^p \right)^{1/p}.$$

We focus on two extreme endpoints: the ℓ_1-norm,

$$||\mu - \eta||_1 = \sum_{\omega} |\mu(\omega) - \eta(\omega)|,$$

and the ℓ_∞-norm, obtained by taking $p \to \infty$:,

$$||\mu - \eta||_\infty = \max_{\omega} |\mu(\omega) - \eta(\omega)|.$$

Intuitively, the ℓ_1-distance measures the *cumulative* changes between our scheduler's distribution and the uniform distribution, while the ℓ_∞-distance measure the *maximum* difference between them on any given interaction. These are two extreme cases – of all distances, ℓ_1-distance is the largest, while ℓ_∞-distance is the smallest. (Our results for ℓ_1 and ℓ_∞ immediately imply results for other ℓ_p-norms, see full version for more details).

3-majority dynamics under a uniform scheduler

It was shown in [14] that under the uniformly random scheduler, if we initially have a gap of $\Omega(\sqrt{n \log n})$ in favor of X, then this process converges, and quickly: w.h.p., within $O(n \log n)$ steps all nodes will have value X (and symmetrically for Y, of course). Although [14] focuses on interactions in triplets, they show that their results can be translated to other models – e.g., to population protocols, where the nodes interact in pairs and have three possible states. We believe that our results here can also be applied to population protocols, similar to [14].

The cost of creating a partition

It is not difficult to see that even a fixed scheduler that is very close to uniform can prevent the system from reaching agreement, by disconnecting one node from the rest of the network: if μ is the distribution where one specific node has probability 0 of appearing in any triplet, and the remaining triplets have uniform probability, then

$$||\mu - U||_1 = 2 \cdot \left(\binom{n-1}{2} / \binom{n}{3} \right) = \frac{1}{\Theta(n)},$$

and

$$||\mu - U||_\infty = \frac{1}{\binom{n}{3}} = \frac{1}{\Theta(n^3)}.$$

More generally, if the initial configuration has a gap of $\Delta < n/4$ in favor of X, the scheduler can choose a set $A \subseteq [n]$ of $2\Delta < n/2$ nodes that have value X, and disconnect the set A from the rest of the system, by setting to 0 the probability of any triplet that involves nodes from both inside and outside A. We refer to this type of scheduler as a *partition-based scheduler*. Disconnecting the set A "flips the majority" in the remainder of the system: the set $[n] \setminus A$ now has a gap of Δ in favor of Y instead, and is likely to converge to Y (which occurs w.h.p when Δ is not too small). The nodes in the set A retain their original value, X, but most nodes in the system are in $[n] \setminus A$ and will convert to Y instead. The ℓ_1-distance incurred by the scheduler is roughly proportional to the size of A; however, due to its nature, the ℓ_∞-distance is still $1/\Theta(n^3)$, dominated by the cost of "deleting" a single triplet.[2]

Our main result in this paper is that *creating a partition is the worst the scheduler can do*: roughly speaking, if the scheduler's distance from uniform does not allow it to disconnect enough nodes to flip the majority in the remainder of the system, then we are guaranteed w.h.p. that most of the system will converge to the initial majority value. However, the details differ significantly between schedulers that are ℓ_1- and ℓ_∞-close to uniform, as we explain next.

Our results

We first consider schedulers that are ℓ_∞-close to uniform, that is, they do not change the probability of any given triplet by much. In this case we have a strong dichotomy: essentially, if the scheduler's distance from uniform does not allow it to disconnect any single node, then convergence to the majority is guaranteed to happen quickly, provided the initial gap is large enough; but if the distance is sufficient to disconnect one node, then we cannot even guarantee that more than 80% of the network converts to the majority value.

▶ **Theorem 1** (Resilience under ℓ_∞-distortion). *The following hold:*

1. *For any $\varepsilon \in [0, 3/4]$ and any $(\infty, \varepsilon/\binom{n}{3})$-close scheduler, if we start at a configuration with a gap of at least $(4\varepsilon n + 16\sqrt{n \log n})/3$ in favor of one value, then with probability $1 - 1/n$ the system reaches consensus on the initial majority value within $O(n \log n)$ steps.*
2. *On the other hand, there is a fixed $(\infty, 1/\binom{n}{3})$-close scheduler S and an initial configuration C_0 where 55% of the nodes have value X, such that in the execution under S starting from C_0, within $O(n \log n)$ steps, the system converges to a configuration where only 20% of the nodes have value X.*

In part (1) of the theorem, the restriction of $\varepsilon \leq 3/4$ is for simplicity; ε can be made arbitrarily close to 1 (but still constant), and the gap required to guarantee fast convergence then approaches $\varepsilon n + \Theta\big(\sqrt{n \log n}\big)$.

For schedulers that are close to uniform in ℓ_1, there is a gap between the distance that can be tolerated in order to guarantee conversion of *most* of the system to the majority, and the distance that can be tolerated in order to guarantee full consensus. We show:

▶ **Theorem 2** (Resilience to ℓ_1-distortions). *The following hold:*

1. *Fix a constant $\alpha \in (0, 1/2)$. If $\varepsilon \leq O(\alpha)$, then under any $(1, 2\varepsilon)$-close scheduler, starting at any configuration with initial gap $\Omega(\varepsilon n + \sqrt{n \log n})$ in favor of one value, w.h.p. within $O(n \log n)$ steps we reach a configuration with gap $\geq (1 - 2\alpha)n$ in favor of the initial majority value, and w.h.p. the gap will never subsequently fall below $(1 - 2\alpha - o(1))n$.*
2. *For any $\Omega\big(\sqrt{n \log n}\big) \leq \varepsilon \leq 1/4$, there exists a fixed $(1, 2\varepsilon)$-close scheduler S and a configuration C_0 with initial gap $O\big(\varepsilon n + \sqrt{n \log n}\big)$ in favor of X, such that in the execution under S starting from C_0, within $O(n \log n)$ steps, w.h.p. the system converges to a configuration where less than $n/2$ nodes hold value X.*
3. *For any $0 \leq \varepsilon < 1/(12n)$, under any $(1, 2\varepsilon)$-close scheduler, starting from any configuration with gap $\Omega(\sqrt{n \log n})$, w.h.p. within $O(n \log n)$ steps we reach consensus on the majority.*

[2] In fact, the exact ℓ_∞-distance of disconnecting one node, $1/\binom{n}{3}$, already allows us to disconnect up to $n/5$ nodes. To disconnect a set A of any size, a slight increase is due, but a distance of $(3 + o(1))/\binom{n}{3}$ suffices.

For example, this shows that we can tolerate an ℓ_1-distance of roughly $O(1/\sqrt{n})$ without asymptotically increasing the initial gap required under the uniform scheduler, $\Omega(\sqrt{n \log n})$. (It is known that without an initial gap of $\Omega(\sqrt{n \log n})$, under the uniform scheduler, the system will still converge w.h.p. to some value, but not necessarily the majority value [7, 14].) If the ℓ_1-distance is bounded away from $1/n$, we get full convergence to the majority, and this distance is tight, as shown by the example of disconnecting a single node.

We also remark that while part (1) of Theorems 1 and 2 can handle schedulers that adaptively change the distribution in every round, the matching lower bound, part (2) of both theorems, uses a partition-based scheduler, where the distribution is always the same. This shows that adaptivity is not essential for the scheduler's power to prevent the system from reaching consensus.

As shown by Theorems 1 and 2, the partition-based scheduler is as bad as any scheduler with a given distance from uniform can be (up to constants). However, is creating a partition the *only* bad thing a scheduler can do, or do there exist "well-connected" schedulers that are close to uniform but still prevent convergence to the majority? In Section 4 we answer this question by showing an example of a scheduler that is "well-connected", in the sense that it has roughly the same weighted edge expansion as the uniform scheduler, but it causes the system to converge to the wrong value (the minority value); we also give an example of a well-connected scheduler that prevents the system from ever reaching a stable configuration.

2 Related Work

The problem of computing the majority has been extensively studied in the population protocol (PP) model [6] and other closely-related models, such as chemical reaction networks (CRNs) [26, 15], under both worst-case and randomized schedulers. A long line of research [6, 7, 24, 17, 21, 4, 22, 23, 1, 12, 2, 11, 10, 9] culminated in a precise characterization [16] of the space-time trade-off for stably computing the exact majority. The optimal protocol of [16] uses $O(\log n)$ memory states, and has a running time of $O(n \log n)$ interactions.

In the finite-memory regime, it is known that exact majority requires at least $\tilde{\Omega}(n^2)$ interactions, and this is achievable using 4 states (i.e., 2 bits of memory) [21, 17]. Furthermore, it is known that exact majority cannot be solved with only 3 states [21]. Approximate majority was first studied in [7], which showed that with an initial gap of $\omega(\sqrt{n} \log n)$, there is a 3-state population protocol that converges to the majority in $O(n \log n)$ interactions. The analysis of the 3-state protocol from [7] was simplified and tightened in [14], by reduction to a one-dimensional random walk: [14] related the population protocol model to chemical reactions with interactions in triplet, which are easier to analyze, and also showed that an initial gap of $\Omega(\sqrt{n \log n})$ is sufficient for convergence. The techniques of [14] are the basis for our upper bounds in this paper.

Most of the work on majority in population protocols and chemical reaction networks assumes either worst-case or uniformly random schedulers, but there are some exceptions. For example, [21] consider a scheduler that is uniformly random, but only over the edges of some fixed connected graph G, rather than all pairs; [21] shows that for this scheduler, the 3-state protocol of [7] might take exponentially long to converge, and could even converge to the wrong value, despite a linear initial gap. Recently, [3] considered the same model, and showed that graphs with high edge-expansion induce a well-behaved scheduler, so that any protocol for the uniform scheduler can be adapted to this model. However, their transformation incurs a super-constant blow-up in memory. Another way to relax the assumption of perfect uniformity is introduced in [25], which studies leader election in a *smoothed analysis* version of population protocols: at each step, with some probability p the next interaction is chosen uniformly at random, and with probability $1-p$ it is chosen adversarially. Some work on majority protocols also considers *Byzantine agents* (e.g., [7, 14]) or noisy communication between agents (e.g., [20, 18, 19]).

3 Preliminaries

3-majority dynamics

As outlined in Section 1, in the 3-majority dynamics we have a system of n nodes, each with an initial opinion (value) of either X or Y. The nodes interact in triplets: when the triplet $\{i,j,k\}$ is scheduled, nodes i,j,k change their opinion to the majority opinion among the three (e.g., $\{\mathsf{X},\mathsf{X},\mathsf{Y}\} \to \{\mathsf{X},\mathsf{X},\mathsf{X}\}$). We assume w.l.o.g. that initially the majority value is X.

A *configuration* C is a mapping $C : [n] \to \{\mathsf{X},\mathsf{Y}\}$ describing the current opinion of each node. We denote by $\#\mathsf{X}(C), \#\mathsf{Y}(C)$ the number of nodes that have opinion X or Y in C, respectively. The *gap* at configuration C is defined as $\Delta(C) = \#\mathsf{X}(C) - \#\mathsf{Y}(C)$. To simplify the notation, when C is clear from the context, we omit it, and write x for $\#\mathsf{X}(C)$, y for $\#\mathsf{Y}(C)$, and Δ for $\Delta(C)$.

Given a set $A \subseteq [n]$ of nodes and $k \geq 1$, we let $\binom{A}{k}$ denote the set of all k-tuples of distinct elements drawn from A (e.g., $\binom{A}{3}$ is the set of all triplets from A).

Three events that feature often in our analysis are the events $\mathcal{X}$, the conversion of a node with value Y to value X (i.e., a triplet with values $\{\mathsf{X},\mathsf{X},\mathsf{Y}\}$); $\mathcal{Y}$, the conversion of a node with value X to value Y; and $\mathcal{P}$, a "productive step", defined by $\mathcal{P} = \mathcal{X} \cup \mathcal{Y}$. Throughout an execution, We say an event holds w.h.p if for any constant $\gamma \geq 1$, the error probability can be reduced to $n^{-\gamma}$ with the cost of constant factors in running time (and possibly in the required initial gap) that only depend on γ.

Schedulers

A *scheduler* is a mapping S from a sequence of configurations representing the system's history so far, to a distribution over triplets in $\binom{[n]}{3}$, from which the next interaction will be drawn. A *run* of a scheduler S starting from an initial configuration C_0 is a sequence of configurations starting at C_0, where each configuration C_i is obtained from C_{i-1} by scheduling an interaction drawn from the distribution $S(C_0,\ldots,C_{i-1})$. A scheduler S is called *fixed* if it always returns the same distribution, for which case we abuse notation and denote that distribution by S; we also use this notation abuse in our upper bounds by writing S instead of $S(C_0,\ldots,C_{i-1})$ to refer to the scheduler's distribution for the current step, when past configurations are already fixed. We let U denote the uniform scheduler, where the next interaction is chosen uniformly at random from $\binom{[n]}{3}$.

A scheduler S is said to be (p,ε)*-close to uniform*, for $p \geq 1$ or $p = \infty$ and $\varepsilon \in [0,2]$, if for every sequence of configurations $C_0,\ldots,C_i$ we have $\|S(C_0,\ldots,C_i) - U\|_p \leq \varepsilon$.

We rely on the following well-known characterization of the ℓ_1-distance between distributions: for any two discrete distributions $\mu,\eta : \Omega \to [0,1]$,

$$\|\mu - \eta\|_1 = 2\max_{E\subseteq\Omega}|\mu(E) - \eta(E)| = 2\sum_{\omega:\mu(\omega)>\eta(\omega)}(\mu(\omega) - \eta(\omega)).$$

In other words, the ℓ_1-distance exactly characterizes the maximum difference between the probabilities assigned by the two distributions to any event.

4 Lower Bounds

In this section we show several lower bounds on the initial gap required to withstand a given distance from the uniform scheduler. We begin with the simple example of a partition-based scheduler, which proves part (2) of Theorems 1 and 2. Then we show two schedulers that do not cause a partition, and in fact leave the network "well-connected", but still prevent it from converging to the majority value: the first causes the network to converge to the minority value w.h.p., and the second prevents it from converging at all. These examples all involve some technical calculations, which can be found in the full version.

4.1 Partition-Based Schedulers

Consider a scheduler S_A that partitions the network into two disconnected sets of nodes, $A \subseteq [n]$ and $B = [n] \setminus A$, by setting to zero the probability of any triplet that involves nodes from both A and B. The remaining triplets are assigned uniform probability:

$$S_A(\{i,j,k\}) = \begin{cases} 0, & \text{if } \{i,j,k\} \notin \binom{A}{3} \cup \binom{B}{3}, \\ \frac{1}{\left|\binom{A}{3} \cup \binom{B}{3}\right|}, & \text{if } \{i,j,k\} \in \binom{A}{3} \cup \binom{B}{3}. \end{cases}$$

We can bound the distance of S_A from the uniform scheduler as follows:

▶ **Lemma 3.** *For any set $A \subseteq [n]$ of size $|A| \leq n/2$, we have $||S_A - U||_1 \leq 12|A|/n$. Furthermore, if $|A| \leq n/5$, then for all sufficiently large n we have $||S_A - U||_\infty \leq \frac{1}{\binom{n}{3}}$.*

We note that for $|A| \leq n/2$ (which allows for any partition), the ℓ_∞-distance incurred by S_A is still at most $(3 + o(1))/\binom{n}{3}$, not much greater than when A is a single node. We omit the distance calculations.

Using the partition-based scheduler, it is easy to see that if the initial configuration has a gap of at most $\varepsilon \cdot n$ in favor of one value, then there is a $(1, O(\varepsilon))$-close to uniform scheduler that causes the system to converge to the minority value: given $\Omega\big(\sqrt{n \log n}\big) \leq \varepsilon \leq n/4$, let C_0 be an initial configuration with an initial gap of $\Delta(C_0) \leq \varepsilon n$ in favor of X. Let $A \subseteq [n]$ be a set of $2\varepsilon n$ nodes with opinion X, and let S_A be the partition-based scheduler defined above. By Lemma 3 we have $||S_A - U||_1 \leq 24\varepsilon$. Under S_A, the nodes in set A always retain their initial opinion of X, since they never communicate with the nodes in $B = [n] \setminus A$; but in the set B there is a large enough majority in favor of Y to guarantee convergence w.h.p. to Y. This proves part (2) of Theorem 2.

With ℓ_∞ the situation is even worse: the ℓ_∞-distance is much more permissive than ℓ_1, in the sense that it allows us to make many small changes while only "paying" for the largest of the changes. In particular, disconnecting a linear-sized set costs as much as disconnecting a single node. Concretely, let C_0 be a configuration where $\#\mathsf{X}(C_0) = 0.55n$. We choose a set A that consists of $0.2n$ nodes with value X. By Lemma 3, S_A is $(\infty, 1/\binom{n}{3})$-close to uniform. However, the set $B = [n] \setminus A$ will converge to Y w.h.p, while the nodes in A will keep their original value, X. Thus, the system converges to a configuration C where $\#\mathsf{Y}(C) = 0.8n$ and $\#\mathsf{X}(C) = 0.2$, which proves part (2) of Theorem 1.

4.2 Well-Connected Schedulers

The previous scheduler caused the system not to agree on the majority value by creating a partition, which prevented some nodes from ever hearing an opinion differing from their own. If we restrict the scheduler so that it is not allowed to create a partition or even come close to doing so, can it still cause the system not to converge to the majority? We show that the answer is *yes*. One way to formalize the degree to which a scheduler is "connected" is to view the scheduler as a *weighted 3-uniform hypergraph*, and consider its weighted edge-expansion, which measures the worst-case ratio between the weight of the hyperedges adjacent to a node set U, and the size of the set U.

Prior work on schedulers with good edge-connectivity

In population protocols, we have a population of n agents that interact in pairs rather than triplets. While much of the work on population protocols assumes either a worst-case scheduler or a uniformly random one, there is prior work on population protocols where the scheduler is restricted to allow only a subset of possible interactions, represented by the edges of an interaction graph G over the agents. The scheduler is then assumed uniformly random over the edges of G. Specifically, for the regime of super-constant agent memory, it was recently shown in [3] that such schedulers are "similar to the uniform scheduler" whenever the graph G has a good edge-connectivity (i.e, it is a good expander). in the sense that any protocol designed for the uniform scheduler over all pairs can be adapted to run correctly under the scheduler S_G that is uniformly random over only the edges of G (at the cost of increasing the agents' memory and the running time of the protocol). However, even disregarding the difference between interaction in pairs and in triplets, these results do not apply to our setting, for two main reasons: first, they do not hold in the constant-memory regime that we focus on here; and second, the scheduler S_G is assumed to be *perfectly uniform* over the edges of G, but in general most schedulers cannot be represented as (or even approximated by) S_G for any graph G. Below we show that there are schedulers (for triplets) that have good edge-expansion but still behave *very* differently from the uniform scheduler.

Edge-connectivity with 3-way interactions

We represent a fixed scheduler S as a weighted 3-uniform hypergraph $G = (V, T)$, where $V = [n]$, $T = \binom{[n]}{3}$ and the weights are the probabilities $S : T \to [0, 1]$ (so that $\sum_{t \in T} S(t) = 1$). For a set of nodes $U \subseteq V$, we define the *boundary* of U to be the set of triplets that include some node from U and some node outside U:

$$\partial U := \{t \in T \ : \ t \cap U \neq \emptyset, t \cap (V \setminus U) \neq \emptyset\}.$$

For a set of triplets $R \subseteq T$, we define its *weight* to be the probability that any triplet from the set is scheduled:

$$wt(R) := \sum_{t \in R} S(r).$$

Using this notation, the (weighted) edge expansion of the scheduler S is defined as:

$$h(S) := \min_{0 < |U| \leq \frac{|V|}{2}} \frac{wt(\partial U)}{|U|},$$

where for any vertex set U, the expression above is the edge-expansion of U, denoted by $h(U)$.

Notice that if U is exactly the set of nodes holding opinion Y, and all the other nodes hold opinion X, then $wt(\partial U)$ is the probability of a productive step, i.e., a step where some node changes its opinion. Thus, good edge-expansion for scheduler S means that for starting at any configuration, under the scheduler S we have a decent probability for a productive interaction, relative to the current number of nodes that hold the minority opinion.

Let us first establish some upper and lower bounds on the possible values of $h(S)$. The partition-based scheduler that we considered above has very bad expansion, $h(S) = 0$ (as does any other partition-based scheduler). A second, slightly less-immediate observation, is the following:

▶ **Observation 4.** *For any fixed scheduler S we have $h(S) \leq 3/n$.*

Proof. Consider only singleton sets, $U = \{i\}$ for some $i \in [n]$. The boundary of U consists exactly of the triplets that contain i, i.e, $\partial U = \{t \in T \ : \ i \in t\}$. If we sum the edge expansion of all such singletons, we have

$$\sum_{i=1}^{n} h(\{i\}) = \sum_{i=1}^{n} \frac{wt(\partial\{i\})}{|\{i\}|} = \sum_{i=1}^{n} \sum_{t \in T : i \in t} S(t) = \sum_{t \in T} 3 \cdot S(t) = 3.$$

Thus, by an averaging argument, there exists some singleton set $\{i\}$ with edge expansion $h(\{i\}) \leq 3/n$, implying that $h(S) \leq h(\{i\}) \leq 3/n$. ◀

As can be expected, the uniform scheduler has very good edge expansion:

▶ **Observation 5.** *The uniform scheduler S has $h(S) \geq 3/(2n)$.*

Proof. By the symmetry of the scheduler, all sets of the same size have exactly the same edge expansion. Let $V' \subseteq V$ be some set of k nodes for $1 \leq k \leq n/2$. Then

$$wt(\partial V) = \frac{k\binom{n-k}{2} + (n-k)\binom{k}{2}}{\binom{n}{3}} = \frac{\frac{k(n-k)(k-1+n-k-1)}{2}}{\frac{n(n-1)(n-2)}{6}} = \frac{3k(n-k)}{n(n-1)}.$$

Normalizing by the set size, we get

$$\frac{wt(\partial V')}{|V'|} = \frac{wt(\partial V')}{k} = \frac{3(n-k)}{n(n-1)}.$$

This is decreasing in k, so taking $k = n/2$, we conclude that $h(U) \geq 3/(2n-2) \geq 3/(2n)$. ◀

Next, we give two "well-connected" schedulers, which both have nearly the same edge-expansion as the uniform scheduler, $\Omega(1/n)$. However, one of schedulers causes the system to converge to the initial minority value, and the other prevents the system from converging at all.

Converging to the wrong value

Our first example is a scheduler that is $(1, O(\varepsilon))$-close to uniform, and is "well-connected" in the sense that it has good weighted edge-expansion, but it causes the network to converge to the wrong value. Let C_0 be an initial configuration where nodes $1, 2$ have opinion Y, and the remaining nodes have a gap of $\Theta(\varepsilon n)$ towards X. We construct a scheduler that always schedules nodes 1 and 2 together whenever they appear, so that the opinion of these two nodes can never flip (any triplet that includes them will already have two Ys). Moreover, the scheduler gives nodes 1,2 "extra influence" over the rest of the nodes, by increasing the probability of triplets of the form $\{1, 2, j\}$ where $j \neq 1, 2$. We prove that this additional influence suffices to push the system away from reaching consensus on X, despite the initial gap of $\Theta(\varepsilon n)$ in favor of X.

More formally, let $m = n-2$, and $\phi = \varepsilon + m/\binom{n}{3}$, and define S be the following scheduler:

- With probability ϕ, choose a uniformly random node $j \in [n] \setminus \{1, 2\}$, and schedule the triplet $\{1, 2, j\}$.
- With probability $1 - \phi$, schedule a uniformly random triplet $\{i, j, k\} \in \binom{[n]\setminus\{1,2\}}{3}$.

The distance of S from the uniform scheduler is given by:

▶ **Lemma 6.** *For any* $\varepsilon \geq 6/n$ *we have* $||S - U||_1 = 2\varepsilon$.

Now fix an initial configuration C_0 where nodes $1, 2$ have opinion Y, and the remaining nodes have a gap of $\varepsilon n/100$ towards X; that is, $\#\mathsf{Y}(C_0)/m = 1/2 - \varepsilon/200$.

We begin by showing that as long as the gap in favor of X is relatively small, then within our productive steps there is a noticeable "pull" towards conversion to Y. Moreover, as long as Y does not already have a very large advantage, there is also a fairly good chance of making a productive step:

▶ **Lemma 7.** *Fix a configuration* C *where* $(1/2 - \varepsilon/100) \leq \#\mathsf{Y}(C)/m \leq 9/10$*, and assume that* $\varepsilon = \Omega(\sqrt{\log n/n})$*. Then for all sufficiently large* n*, we have*

$$\Pr_S[\mathcal{X}] + \Pr_S[\mathcal{Y}] \geq \frac{1}{25}, \qquad \text{and} \qquad \Pr_S[\mathcal{Y}] - \Pr_S[\mathcal{X}] \geq \frac{\varepsilon}{25}.$$

Recall that the network starts with a gap of $\Theta(\varepsilon n)$ in favor of X. Next we show that because of the bias in favor of converting to Y, w.h.p., the network will *never* reach a gap in favor of X that is twice the initial gap:

▶ **Lemma 8.** *Fix a configuration* C_0 *with* $\#\mathsf{Y}(C_0)/m = 1/2 - \varepsilon/200$*, and assume that* $\varepsilon \geq \Omega(\sqrt{\log n/n})$*. Then under the scheduler* S*, w.h.p the system never reaches a configuration* C *with* $\#\mathsf{Y}(C)/m \leq 1/2 - \varepsilon/100$*.*

Combining the two lemmas, we get the following.

▶ **Lemma 9.** *Fix a configuration* C_0 *with* $\#\mathsf{Y}(C_0)/m = 1/2 - \varepsilon/200$*, and assume that* $\varepsilon \geq \Omega(\sqrt{\log n/n})$*. Then under the scheduler* S*, within* $500n$ *steps, w.h.p the system reaches a configuration* C_1 *where* $\#\mathsf{Y}(C_1)/m \geq 1/2 + 5\varepsilon$*.*

Starting from configuration C_1, we apply the first part of Theorem 2 to conclude that the system converges w.h.p. to Y instead of to X, within a total of $O(n \log n)$ steps overall.

The connectivity of the scheduler we defined here is not far from that of the uniform scheduler. This is implied by the following claim, which requires some calculations:

▷ Claim 10. For any set $U \subseteq V$ of size at most $n/2$, the edge expansion of U is at least

$$\frac{wt(\partial U)}{|U|} \geq \Theta(\min\{\phi, 1/m\}).$$

We use the scheduler with $\varepsilon = \Omega(1/n)$ and thus $\phi = \Omega(1/n)$; also recall that $m = n - 2$. Thus, the edge expansion of the scheduler is $\Theta(1/n)$.

Never reaching a stable configuration

Is convergence to some stable configuration guaranteed under any fixed scheduler? We show that if the scheduler is far enough from uniform, the answer is *no*: consider an initial configuration where nodes $1, 2$ have value X, and the other nodes have value Y, and let S be a scheduler that chooses a uniformly random node $j \in \{5, \ldots, n\}$, and schedules w.p. $1/2$ the triplet $\{1, 2, j\}$ and w.p. $1/2$ the triplet $\{3, 4, j\}$. Since nodes $1, 2$ and $3, 4$ are always scheduled together, they always retain their initial value, X or Y respectively. We refer to these two pairs as *fixed pairs*, and to the remaining nodes as *free nodes*. Because each free node $j \in \{5, \ldots, n\}$ interacts with nodes $1, 2$ and with nodes $3, 4$ at least once every $O(n \log n)$

rounds (w.h.p.), the value of each of these nodes swings between X and Y repeatedly. We can never reach convergence in the first place, because the fixed pairs cannot change their opinions, but in fact, most of the time the opinions in the system are roughly evenly split between X and Y. This is true as in each interaction the chosen free node j simply resets its opinion to either X or Y with even probability, depending on the fixed pair used. The outcome of the interaction does not depend on the previous values held by j or any other free node.

Next we calculate the edge expansion of this scheduler. Let $m = n - 4$ be the number of free nodes. The scheduler allows only $2m$ triplets to be scheduled: each free node has one triplet with each fixed pair, $1, 2$ and $3, 4$. All triplets t have the same probability, $S(t) = 1/(2m)$. We bound the edge-expansion of any set U of at most $n/2$ nodes.

First, consider sets U that "split" a fixed pair: for either the pair $1, 2$ or the pair $3, 4$, the set U includes one node from the pair but not the other. The boundary ∂U then contains all m triplets that contain the pair split by U, and therefore $wt(\partial u) \geq m/(2m) = 1/2$. For sets U of size $|U| \leq n/2$, the edge-expansion of U is thus at least $wt(\partial U)/|U| \geq (1/2)/(n/2) = 1/n$.

Next, consider the case where no fixed pair is split by U, but U contains one of the fixed pairs and not the other. For each free node $j \in V$ (either in U or in $V \setminus U$) exactly one triplet contributes to the boundary, the one containing j and the fixed pair on the other side (either $V \setminus U$ or U), and therefore $wt(\partial U) = m/(2m) = 1/2$, and same as before, the edge-expansion of U is at least $1/n$.

The last remaining case is sets U that either include both fixed pairs, or neither. Assume first that U includes neither fixed pair, that is, it contains only free nodes. The boundary of U includes exactly $2|U|$ triplets: each free node in U contributes 2 triplets, one for each fixed pair (outside U), and therefore $wt(\partial U) = 2|U|/(2m) = |U|/m$. Thus, the edge-expansion is $h(U) = |U|/m/(|U|) = 1/m \geq 1/n$.

Now suppose U includes both fixed pairs. Then the complement, $V \setminus U$, includes neither, and by the same argument as before $wt(\partial U) = wt(\partial(V \setminus U)) = 2|V \setminus U|/(2m) = |V \setminus U|/m$. We recall that $|U| \leq n/2 \leq |V \setminus U|$. Therefore,

$$h(U) = \frac{wt(\partial U)}{|U|} = \frac{|V \setminus U|/m}{|U|} \geq \frac{|U|/m}{|U|} = 1/m > 1/n.$$

In all cases above, the expansion is at least $1/n$, and thus $h(S) \geq 1/n$.

5 Convergence Under Bounded Distance from the Uniform Scheduler

Our upper bounds that show convergence under bounded ℓ_∞- or ℓ_1-distance broadly follow the outline of the analysis for the uniform scheduler, from [14]. The analysis has three phases: in Phase I, it is shown that an initial gap of $\sqrt{n \log n}$ is likely to increase consistently until a constant-factor gap (e.g., 99%) is reached. In Phase II, the small constant fraction of nodes that hold the minority opinion holders decreases, until almost none are left, at most $O(\log n)$. In phase III the network "chases" after the few "straggling nodes" that are left, which takes time only due to their scarcity.

In our setting, the main obstacle is to understand the ways in which a scheduler that deviates from the uniform distribution can cause problems. These can be put into two categories.

Distorting high-probability events. In Phase I, while the system is still somewhat evenly divided between X and Y, nodes change opinions very frequently: triplets with opinions $\{X, X, Y\}$ or $\{Y, Y, X\}$ are very likely, but because we have a small gap in favor of X, triplets with opinions $\{\mathsf{X}, \mathsf{X}, \mathsf{Y}\}$ have a small but significant advantage: the events $\mathcal{X}$ and $\mathcal{Y}$ both have probability $\Omega(1)$, but the difference between them is $\Pr[\mathcal{X}] - \Pr[\mathcal{Y}] = \Theta(\Delta/n)$, where Δ is the current gap between X and Y. Over time, this bias towards X increases the gap, until a large constant fraction of the system is converted to X.

A scheduler that is very close to uniform cannot greatly distort the probability of a large event: if the scheduler is $(1, \varepsilon)$-close to uniform, it can only alter the probability of the events $\mathcal{X}, \mathcal{Y}$ by at most an additive $\pm\varepsilon$, and if the scheduler is $(\infty, \varepsilon/\binom{n}{3})$-close to uniform, it can only alter these probabilities by a multiplicative factor of $(1 \pm \varepsilon)$. This distortion is not significant when $\mathcal{X}, \mathcal{Y}$ are $\Omega(1)$. However, what the scheduler *can* significantly affect is the *gap* between the events: since $\Pr[\mathcal{X}] - \Pr[\mathcal{Y}] = \Theta(\Delta/n)$, and we are working with $\varepsilon = \Theta(\Delta/n)$, the scheduler could theoretically wipe out the bias in favor of X, and cause the system to tend towards Y (as demonstrated by the examples in Section 4). Nevertheless, we are able to show that when ε is smaller than Δ/n by a sufficiently small constant factor, enough bias towards X remains that the system still trends strongly towards X, allowing Phase I of the analysis to go through. The analysis is fairly similar for both ℓ_1 and ℓ_∞, although the exact details differ.

Distorting rare events. In Phases II and III of the analysis, the majority is already well-established, and we wish to show that the few remaining Y eventually convert to X. However, at this point the event $\mathcal{X}$ becomes less and less likely, because it requires a triplet with values $\{\mathsf{X}, \mathsf{X}, \mathsf{Y}\}$, and very few Ys remain. The event $\mathcal{Y}$, a triplet $\{\mathsf{Y}, \mathsf{Y}, \mathsf{X}\}$, is significantly less likely, and in Phase III it can practically be neglected; our concern is that most of the time the triplet scheduled will have values $\{\mathsf{X}, \mathsf{X}, \mathsf{X}\}$, an "unproductive" step.

Since we are now working with events that are potentially very small (i.e., have low probability), the distortion of the scheduler can directly affect the probability of these events in a significant manner. Here, the effect of ℓ_∞-distortion is very different from that of ℓ_1-distortion.

An ℓ_∞-distance of $\varepsilon/\binom{n}{3}$ translates to a multiplicative factor of $(1 \pm \varepsilon)$ in the probability of any event. This multiplicative distortion is fairly well-behaved, even for small events, and it allows the analysis of Phases II and III to go through even when ε is arbitrarily close to 1 (recall from Section 1 that an ℓ_∞-distance of $1/\binom{n}{3}$ is enough to disconnect a node).

In ℓ_1-distance, the situation is more difficult, because a small additive distortion can completely erase a small event; we cannot guarantee full convergence even when $\varepsilon = 1/n$, as shown by the example that disconnects one node. (Phase I still goes through, allowing us to prove that a large constant fraction of the system will agree on the majority value.) However, when $\varepsilon < 1/n$, we are able to cope with the additive distortion and prove that we still get full convergence with high probability.

6 Convergence Under Bounded ℓ_∞-Distortion

In the sequel we use x, y to denote $\#\mathsf{X}(C), \#\mathsf{Y}(C)$ when the configuration C is clear from the context, and we similarly conflate the scheduler S with the distribution $S(C_0, \ldots, C_i)$ that it chooses at $C_0, \ldots, C_i$. Some proofs in this section are deferred to the full version.

A key lemma that is repeatedly used in our analysis is the following, which bounds how much an $(\infty, \varepsilon/\binom{n}{3})$-close scheduler can influence the probabilities of important events in the system. Roughly speaking, we show that the scheduler's effect on the events $\mathcal{X}, \mathcal{Y}$ is at most a multiplicative $(1 \pm \varepsilon)$ (as claimed in the previous section), and that the scheduler is also unable to greatly distort the ratio between the probabilities of these events.

▷ Claim 11 (Probability bounds for ℓ_∞). Fix $0 \le \varepsilon < 1$, and a configuration with $y \le n/2 \le x$. Then for any scheduler S with $||S - U||_\infty \le \varepsilon / \binom{n}{3}$,

1. The probability of converting a Y to an X is bounded from below by: $\Pr_S[\mathcal{X}] \ge 3(1-\varepsilon)\frac{x^2 y}{n^2(n-1)}$.
2. The probability of converting a X to an Y is bounded from above by: $\Pr_S[\mathcal{Y}] \le 3(1+\varepsilon)\frac{xy^2}{n^2(n-1)}$.
3. The probability of a productive step (converting an X to a Y, or vice-versa) is bounded from below by: $\Pr_S[\mathcal{P}] \ge 3(1-\varepsilon)\frac{xy}{n(n-1)}$.
4. For the ratio between the probability of converting a Y to an X and the probability of the reverse, we have: $\frac{\Pr_S[Y]}{\Pr_S[X]} \le \frac{(1+\varepsilon)y}{(1-\varepsilon)(n-y)}$.
5. If we further assume $\Delta = x - y \ge \varepsilon n$, then the probability that a productive step converts an X to a Y is bounded from below by: $\Pr_S[\mathcal{X} \mid \mathcal{P}] \ge \frac{1}{2} + \frac{\Delta - \varepsilon n}{2n}$.

Throughout the analysis, we repeatedly argue that after each stage, there is only a very small chance that we will *ever* regress significantly from the current gap. This allows us to treat our progress as essentially "irreversible". The next lemma captures the trade-off between the current gap, and how far back we might regress, after discounting negligible events. As in [14], it is proven using a *gambler's ruin argument*: the probability of taking a step backwards (event $\mathcal{Y}$) is non-trivially smaller than that of taking a step forwards (event $\mathcal{X}$), and using the bounded ratio between these probabilities that we proved in Claim 11, we get the following.

▷ Claim 12. Fix $0 \le \varepsilon < 1$, and a scheduler S which is $(\infty, \varepsilon/\binom{n}{3})$-close. Fix any margin $d \in \mathbb{N}$, any configuration with initial gap $\Delta_0 \ge \varepsilon n + 3d$ (alternatively, $y_0 \le (n - \varepsilon n - 3d)/2$). Then the probability of *any* future configuration reaching a gap of $\Delta = \Delta_0 - 2d$ (alternatively, $y = y_0 + d$) is at most $\exp\left(-\frac{2(\Delta_0 - 2d - \varepsilon n)d}{(1-\varepsilon)n}\right)$.

We now outline the three phases of the proof, which successively show that gap grows and grows, until consensus on the majority is established.

Phase I

In the first phase, our goal is to start from the initial gap and show that the system reaches a large gap of $\Delta = 998n/1000$ (that is, $y = n/1000$). We divide the phase into stages: in each stage, we start with some gap Δ_0, and the stage ends as soon as we double the initial gap, reaching a configuration with gap $\Delta_{\max} = 2\Delta_0$. We claim the following:

▷ Claim 13 (Phase I stage). Fix any $0 \le \varepsilon < 3/4$ and a scheduler S which is $(\infty, \varepsilon/\binom{n}{3})$-close, and set $d = \sqrt{\gamma n \log(n)}$. Fix a starting gap Δ_0 such that $(4\varepsilon n + 16d)/3 \le \Delta_0 \le 998n/1000$, set a target gap $\Delta_{\max} = \min\{2\Delta_0, 998n/1000\}$, and consider an execution of the majority protocol starting with any configuration of gap Δ_0. The probability that we do not reach a configuration with gap $\Delta_{\max}$ within $t = 2000n/(1-\varepsilon)$ steps is at most $3n^{-2\gamma}$.

Proof. We consider three bad events:

1. Reaching a configuration with a gap significantly smaller than Δ_0,
2. Having too few productive steps within the t steps we consider,
3. Within the productive steps, having too few steps that go in the right direction (X).

For the first bad event, we use Claim 12 to bound its probability: the probability of *ever* reaching a configuration with gap of $\Delta_{\min} = \Delta_0 - 2d$ is at most $\exp\left(-\frac{2(\Delta_0 - 2d - \varepsilon n)d}{(1-\varepsilon)n}\right) \le \exp\left(-\frac{2d^2}{n}\right) = \exp(-2\gamma \log(n)) = n^{-2\gamma}$, where the inequality uses the fact that $\Delta_0 \ge \varepsilon n + 3d$ and $\varepsilon > 0$. In the sequel, we may therefore assume that the gap stays above $\Delta_{\min}$ (and therefore above εn) throughout the stage.

The second bad event is having too few productive steps. Since Phase I only aims to reduce the minority to $y = n/1000$, during the entire phase, there is a fairly good probability of having a productive step: as long as $y \geq n/1000$ we have $xy = y(n-y) \geq n^2/2000$, and using the bound from Claim 11,

$$\Pr[\mathcal{P}] \geq 3(1-\varepsilon)\left(\frac{xy}{n^2}\right) \geq \frac{3(1-\varepsilon)}{2000}.$$

Denote by p the number of productive interactions within $t = 2000n/(1-\varepsilon)$ steps. The bound above implies $\mathbb{E}[p] \geq 3n$. By Chernoff, $\Pr[p < 2n] \leq n^{-2\gamma}$.

Lastly, we bound the number of steps in each direction, within the $2n$ productive steps that are guaranteed to us with high probability. Claim 11 gives us a bound on the probability that we create a new X given a productive step, and the bound is monotonic in the current gap; thus, throughout the stage,

$$\Pr[\mathcal{X} \mid \mathcal{P}] \geq \frac{1}{2} + \frac{\Delta_{\min} - \varepsilon n}{2n}.$$

Denote by r (resp., ℓ) the number of interactions producing X (resp., Y) during the first $2n$ productive steps. Taking $r \geq n + \Delta_0/4$ suffices to guarantee the gap we need.

Finally, our lower bound for $\mathbb{E}[r]$ is indeed larger than $n + \Delta_0/4$, and so by Chernoff, $\Pr[r < n + \Delta_0/4] \leq n^{-2\gamma}$. All together, if none of the three bad events happens, then within $t = 2000n/(1-\varepsilon)$ steps, we get at least $2n$ *productive* steps, which give us a surplus of $\Delta_0/4$ interactions producing X as opposed to producing Y. This is enough to reach our target gap of $\Delta_{\max}$. The probability of any of the bad events occuring is at most $3n^{-2\gamma}$. ◁

The progress made in Phase I is summarized as follows:

▶ **Corollary 14** (Phase I). *Fix $0 \leq \varepsilon < 3/4$ and a scheduler S which is $(\infty, \varepsilon/\binom{n}{3})$-close and $d = \sqrt{\gamma n \log(n)}$. Starting at any configuration with gap $\Delta_0 \geq (4\varepsilon n + 16d)/3$, the probability that within $2000n\log(n)/(1-\varepsilon)$ steps we do not reach a configuration with gap $998n/1000$ is at most $n^{-(\gamma+1)}$.*

Phase II

In the second phase, our goal is to take the gap of $998n/1000$ and keep increasing it towards near-full consensus. In this phase we think in terms of y, the number of minority nodes remaining, as this number determines the probability of a productive interaction. We start Phase II with $y = n/1000$, and our goal is to reduce y to $20\gamma \log n$.

We again divide the phase into stages, each starting at a configuration where some number y_0 of nodes have value Y and ending as soon as we reach a configuration where at most $y_{\min} = y_0/2$ have opinion Y. We claim the following:

▷ Claim 15 (Phase II stage). Fix $0 \leq \varepsilon < 3/4$ and a scheduler S which is $(\infty, \varepsilon/\binom{n}{3})$-close, and set $d = 20\gamma\log(n)$. Fix a starting minority count y_0 such that $2d \leq y_0 \leq n/1000$, set a target value $y_{\min} = y_0/2$, and consider an execution of the majority protocol starting at any configuration with y_0 minority nodes. The probability that we do not reach a configuration with $\leq y_{\min}$ minority nodes within $t = 16n/(1-\varepsilon)$ steps is at most $3n^{-2\gamma}$.

To prove this claim, we first bound the probability of ever falling too far back, as we did in Claim 13. Next, we prove that w.h.p., within t steps, we have "enough" interactions that convert Y to X (at least $c_1 y_0$ such interactions, for some constant c_1), and "not too many" interactions that produce Y (at most $c_2 y_0$ such interactions, for some constant c_2), so that after at most t steps we reduce the number of Ys to at most $y_0/2$.

We conclude Phase II:

▶ **Corollary 16** (Phase II). *Fix $0 \le \varepsilon < 3/4$ and a scheduler S which is $(\infty, \varepsilon/\binom{n}{3})$-close. Starting at any configuration with minority count $y \le n/1000$, the probability that within $16n\log(n)/(1-\varepsilon)$ steps we do not reach a configuration with $y = 20\gamma\log(n)$ is at most $n^{-(\gamma+1)}$.*

Phase III

For the last phase we do not require many stages. We begin this phase with at most $y_0 = 20\gamma\log(n)$ that have opinion Y, and aim to reach full consensus ($y_{\min} = 0$) within a single stage.

▷ Claim 17 (Phase III stage). Fix $0 \le \varepsilon < 3/4$ and a scheduler S which is $(\infty, \varepsilon/\binom{n}{3})$-close, and set $d = 20\gamma\log(n)$. Fix a starting minority count $y_0 \le 40\gamma\log(n) = 2d$ and consider an execution of the majority protocol starting with any configuration with y_0 minority nodes. The probability that we does not reach full consensus on X within $t = 80nd/(1-\varepsilon)$ steps is at most $3n^{-2\gamma}$.

At this very late point in the execution, the probability of a productive step can be very small: it could be that only a single Y remains, and the probability of interacting with it is roughly $(1-\varepsilon)/n$. Nevertheless, over $t = \Omega(n\log n)$ steps we will interact with all the remaining Ys (w.h.p.). When such an interaction occurs, the chances that it converts a Y to an X are overwhelming, because the number of remaining Ys is extremely small, and converting an X to a Y requires *two* Ys. Thus we are able to show that with very high probability, after t steps, no Ys remain.

Combining the three phases (Corollaries 14 and 16 and Claim 17) yields part (1) of Theorem 1.

7 Convergence Under Bounded ℓ_1-Distortion

Any scheduler that is $(\varepsilon/\binom{n}{3})$-close in ℓ_∞ is also ε-close in ℓ_1, but not vice-versa. In this section we briefly sketch our proof that the 3-majority dynamics also converges under a scheduler that is close to uniform in ℓ_1, but the scheduler must be *very* close to uniform, or else we can only guarantee that a large constant fraction will agree on the majority. All proofs in this section are deferred to the full version.

We begin again by bounding the distortion that the scheduler can cause to the probabilities of the important events we consider:

▷ Claim 18 (Probability bounds for ℓ_1). Fix $0 \le \varepsilon < 1$, and a configuration with $y \le n/2 \le x$. Then for any scheduler S which is $(1, 2\varepsilon)$-close,

1. The probability of converting a Y to an X is bounded from below by:
$$\Pr_S[\mathcal{X}] \ge 3\frac{x^2 y}{n^2(n-1)} - \varepsilon.$$
2. The probability of converting a Y to an X is bounded from above by:
$$\Pr_S[\mathcal{Y}] \le 3\frac{xy^2}{n^2(n-1)} + \varepsilon.$$
3. The probability of a productive step (converting an X to a Y, or vice-versa) is bounded from below by:
$$\Pr_S[\mathcal{P}] \ge 3\frac{xy}{n(n-1)} - \varepsilon.$$

4. If $x \in [\alpha n, (1-\alpha)n]$ for $\alpha \in (0, 1/2]$, and we further assume $\Delta \geq 4\varepsilon n/(3\alpha)$, then for the ratio between the probability of converting a Y to an X and the probability of the reverse, we have:

$$\frac{\Pr[\mathcal{Y}]}{\Pr[\mathcal{X}]} \leq 1 - \frac{\Delta - \frac{4\varepsilon n}{3\alpha}}{n}.$$

5. If $x \in [\alpha n, (1-\alpha)n]$ for $\alpha \in (0, 1/2]$, then the probability that a productive step converts an X to a Y is bounded from below by:

$$\Pr[\mathcal{X} \mid \mathcal{P}] \geq \frac{1}{2} + \frac{\Delta - \frac{4\varepsilon n}{3\alpha}}{2n}.$$

Note that these bounds are additive in $\pm\varepsilon$, unlike the multiplicative bounds in the corresponding claim for the ℓ_∞-norm (Claim 11). Also, here we are only able to say a productive step is more likely to convert an X to a Y *when the minority opinion still holds a linear number of nodes.* This is perhaps counter-intuitive: with a larger gap, one might expect a larger inclination towards X. However, as explained in Section 5, both events $\mathcal{X}, \mathcal{Y}$ hinge on an interaction including one of the few Y nodes that remain, and for these the additive distortion can be destructive.

The convergence analysis for bounded ℓ_1 distance is divided into three phases, with the similar guidelines as in the ℓ_∞-case; However, in the ℓ_1 regime, events with small probability are more susceptible to the distortion of the scheduler: once $\Theta(\varepsilon n)$ nodes with opinion Y remain, a malicious scheduler can simply disconnect them, preventing full convergences. For Phase II and Phase III, we prove that if $\varepsilon = O(1/n)$, even events with small probability can endure the distortion. Not only that, they even occur often enough to guarantee the same (asymptotic) running time as the uniform scheduler.

Small chance of regression

As before, first we establish that once a certain gap is reached, there is only a small probability of *ever* regressing too much. A parallel to Claim 12 can be shown for ℓ_1, but with one major difference: such a guarantee now depends on the value α such that *at least* αn minority nodes are still present. This allows to control the additive distortion:

▷ Claim 19. Fix $0 < \alpha < 1/2$. Fix $0 \leq \varepsilon < 1$, and a scheduler S which is $(1, 2\varepsilon)$-close. Fix any margin $d \in \mathbb{N}$, and any configuration with gap $4\varepsilon n/(3\alpha) + 3d \leq \Delta_0 \leq n(1-2\alpha)$ (alternatively, $\alpha n \leq y_0 \leq (n - 4\varepsilon n/(3\alpha) - 3d)/2$). Then the probability of any future configuration reaching a gap $\Delta = \Delta_0 - 2d$ (alternatively, $y = y_0 + d$) is at most

$$\exp\left(-\frac{2(\Delta_0 - 2d - \frac{4\varepsilon}{3\alpha}n)d}{n}\right).$$

This bound gets worse for smaller values of α, and it is carefully applied with different values of α to minimize the error accumulated along the way. At every stage in the process that aims to reduce the number of minority nodes to y, Claim 19 is applied using $\alpha = y/n$.

Phase I, revisited

We have seen that with $\varepsilon \geq \Theta(1/n)$ full convergence is not achievable. However, one might still hope to get a large agreement for the majority value. Indeed, we show that in essence disconnecting $\Theta(\varepsilon n)$ is the worst a malicious scheduler can do. We switch the original goal of

Phase I: instead of converting all but $n/1000$ nodes to the majority, it now aims to convert all but αn nodes (where $\alpha \geq \Theta(\varepsilon)$ can now be a much smaller constant).

With the help of part (5) in Claim 18, we show:

▷ Claim 20 (Phase I stage, ℓ_1). Fix $0 < \alpha < 1/2$. Fix $0 \leq \varepsilon < 9\alpha/16$, and a scheduler S which is $(1, 2\varepsilon)$-close. set $d = \sqrt{\gamma n \log(n)}$. Fix a starting gap Δ_0 such that $16\varepsilon n/(9\alpha) + 16d/3 \leq \Delta_0 \leq (1-2\alpha)n$, set a target gap $\Delta_{\max} = \min\{2\Delta_0, (1-2\alpha)n\}$, and consider an execution of the majority protocol starting with any configuration of gap Δ_0. The probability it does not reach a configuration with gap $\Delta_{\max}$ within $t = 4n/\alpha$ steps is at most $3n^{-2\gamma}$.

Repeatedly applying this claim, along Claim 19, proves part (1) of Theorem 2.

Reaching consensus

Whenever $\varepsilon \leq O(1/n)$, we simply plug in $\alpha = 1/3000 \geq \varepsilon$ to complete Phase I with $n/3000$ minority nodes.

Phase II is composed of many stages. Applying Claim 19 with the correct value of α, we get:

▷ Claim 21 (Phase II stage, ℓ_1). Fix $0 \leq \varepsilon \leq 1/(12n)$ and a scheduler S which is $(1, 2\varepsilon)$-close, and set $d = 2\gamma \log(n)$. Fix a starting minority count y_0 such that $d \leq y_0 \leq n/2304$, set a target value $y_{\min} = \max\{y_0/2, d\}$, and consider an execution of the majority protocol starting with any configuration with y_0 minority nodes. The probability it does not reach a configuration with $y_{\min}$ minority nodes within $t = 64n$ steps is at most $3n^{-2\gamma}$.

Applying this claim repeatedly, we get the following version of Phase II for ℓ_1:

► **Corollary 22** (Phase II, ℓ_1). *Fix $0 \leq \varepsilon < 1/(12n)$ and a scheduler S which is $(1, 2\varepsilon)$-close. Starting at any configuration with minority count $y \leq n/2304$, the probability that within $64n \log(n)$ steps we do not reach a configuration with $y = 2\gamma \log(n)$ is at most $n^{-(\gamma+1)}$.*

Finally, Phase III uses a single stage to complete the process:

▷ Claim 23 (Phase III, ℓ_1). Fix $0 \leq \varepsilon < 1/(12n)$ and a scheduler S which is $(1, 2\varepsilon)$-close, and set $d = 2\gamma \log(n)$. Fix a starting minority count $y_0 = d$, set a target value $y_{\min} = 0$, and consider an execution of the majority protocol starting with any configuration with y_0 minority nodes. The probability it does not reach a configuration with $y_{\min}$ minority nodes within $t = 64\gamma n \log n$ steps is at most $3n^{-2\gamma}$

Combining the three phases (part (1) of Theorem 2 with $\alpha = 1/3000$, along with Corollary 22 and Claim 23) yields part (3) of Theorem 2.

8 Conclusion

In this paper we initiate the study of majority dynamics under schedulers that are *close* to uniform, but may have some non-negligible distance. We focused on the 3-majority dynamics from [14], and showed that this natural process is fairly robust, and can essentially tolerate any distance short of the distance required to partition the network. While we focused here on 3-majority dynamics, the techniques used to analyze them apply also to population protocols (e.g., the approximate majority protocol from [7]) and to other types of chemical reaction networks, and so we believe that our techniques will have more general applicability. We hope that our work can serve as a template for analyzing other protocols and dynamics, leading to a better understanding of the robustness of population protocols and related models, and to the development of robust algorithms for these models.

References

1 Dan Alistarh, James Aspnes, David Eisenstat, Rati Gelashvili, and Ronald L Rivest. Time-space trade-offs in population protocols. In *Proceedings of the twenty-eighth annual ACM-SIAM symposium on discrete algorithms*, pages 2560–2579. SIAM, 2017.

2 Dan Alistarh, James Aspnes, and Rati Gelashvili. Space-optimal majority in population protocols. In *SODA*, pages 2221–2239, 2018.

3 Dan Alistarh, Rati Gelashvili, and Joel Rybicki. Fast graphical population protocols. In *25th International Conference on Principles of Distributed Systems (OPODIS 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

4 Dan Alistarh, Rati Gelashvili, and Milan Vojnović. Fast and exact majority in population protocols. In *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing*, pages 47–56, 2015.

5 Talley Amir, James Aspnes, David Doty, Mahsa Eftekhari, and Eric Severson. Message complexity of population protocols. In *34th International Symposium on Distributed Computing (DISC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

6 Dana Angluin, James Aspnes, Zoë Diamadi, Michael J Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed computing*, 18(4):235–253, 2006.

7 Dana Angluin, James Aspnes, and David Eisenstat. A simple population protocol for fast robust approximate majority. *Distributed Computing*, 21(2):87–102, 2008.

8 James Aspnes and Eric Ruppert. An introduction to population protocols. *Middleware for Network Eccentric and Mobile Applications*, pages 97–120, 2009.

9 Stav Ben-Nun, Tsvi Kopelowitz, Matan Kraus, and Ely Porat. An o (log3/2 n) parallel time population protocol for majority with o (log n) states. In *Proceedings of the 39th Symposium on Principles of Distributed Computing*, pages 191–199, 2020.

10 Petra Berenbrink, Robert Elsässer, Tom Friedetzky, Dominik Kaaser, Peter Kling, and Tomasz Radzik. A population protocol for exact majority with o (log5/3 n) stabilization time and theta (log n) states. In *32nd International Symposium on Distributed Computing (DISC 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

11 Petra Berenbrink, Robert Elsässer, Tom Friedetzky, Dominik Kaaser, Peter Kling, and Tomasz Radzik. Time-space trade-offs in population protocols for the majority problem. *Distributed Computing*, 34(2):91–111, 2021.

12 Andreas Bilke, Colin Cooper, Robert Elsässer, and Tomasz Radzik. Brief announcement: Population protocols for leader election and exact majority with o (log2 n) states and o (log2 n) convergence time. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, pages 451–453, 2017.

13 Yuan-Jyue Chen, Neil Dalchau, Niranjan Srinivas, Andrew Phillips, Luca Cardelli, David Soloveichik, and Georg Seelig. Programmable chemical controllers made from dna. *Nature nanotechnology*, 8(10):755–762, 2013.

14 Anne Condon, Monir Hajiaghayi, David Kirkpatrick, and Ján Maňuch. Approximate majority analyses using tri-molecular chemical reaction networks. *Natural Computing*, 19(1):249–270, 2020.

15 Matthew Cook, David Soloveichik, Erik Winfree, and Jehoshua Bruck. Programmability of chemical reaction networks. In *Algorithmic bioprocesses*, pages 543–584. Springer, 2009.

16 David Doty, Mahsa Eftekhari, Leszek Gasieniec, Eric Severson, Przemyslaw Uznański, and Grzegorz Stachowiak. A time and space optimal stable population protocol solving exact majority. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1044–1055. IEEE, 2022.

17 Moez Draief and Milan Vojnovic. Convergence speed of binary interval consensus. *SIAM Journal on Control and Optimization*, 50(3):1087–1109, 2012.

18 Francesco d'Amore, Andrea Clementi, and Emanuele Natale. Phase transition of a non-linear opinion dynamics with noisy interactions. In *International Colloquium on Structural Information and Communication Complexity*, pages 255–272. Springer, 2020.

19 Francesco d'Amore and Isabella Ziccardi. Phase transition of the 3-majority dynamics with uniform communication noise. In *International Colloquium on Structural Information and Communication Complexity*, pages 98–115. Springer, 2022.

20 Ofer Feinerman, Bernhard Haeupler, and Amos Korman. Breathe before speaking: Efficient information dissemination despite noisy, limited and anonymous communication. In *PODC'14*, pages 114–123, 2014.

21 George B. Mertzios, Sotiris E. Nikoletseas, Christoforos L. Raptopoulos, and Paul G. Spirakis. Determining majority in networks with local interactions and very small local memory. *Distributed Comput.*, 30(1):1–16, 2017.

22 Yves Mocquard, Emmanuelle Anceaume, James Aspnes, Yann Busnel, and Bruno Sericola. Counting with population protocols. In *2015 IEEE 14th International Symposium on Network Computing and Applications*, pages 35–42. IEEE, 2015.

23 Yves Mocquard, Emmanuelle Anceaume, and Bruno Sericola. Optimal proportion computation with population protocols. In *2016 IEEE 15th International Symposium on Network Computing and Applications (NCA)*, pages 216–223. IEEE, 2016.

24 Etienne Perron, Dinkar Vasudevan, and Milan Vojnovic. Using three states for binary consensus on complete graphs. In *IEEE INFOCOM 2009*, pages 2527–2535. IEEE, 2009.

25 Gregory Schwartzman and Yuichi Sudo. Smoothed analysis of population protocols. In *DISC*, volume 209 of *LIPIcs*, pages 34:1–34:19, 2021.

26 David Soloveichik, Matthew Cook, Erik Winfree, and Jehoshua Bruck. Computation with finite stochastic chemical reaction networks. *natural computing*, 7(4):615–633, 2008.

27 Niranjan Srinivas, James Parkin, Georg Seelig, Erik Winfree, and David Soloveichik. Enzyme-free nucleic acid dynamical systems. *Science*, 358(6369):eaal2052, 2017.