



# Near-Optimal Leader Election in Population Protocols on Graphs

Dan Alistarh<sup>1</sup> · Joel Rybicki<sup>2</sup> · Sasha Voitovych<sup>3</sup>

Received: 19 December 2022 / Accepted: 20 May 2025 / Published online: 27 June 2025  
© The Author(s) 2025

## Abstract

In the *stochastic population protocol* model, we are given a connected graph with  $n$  nodes, and in every time step, a scheduler samples an edge of the graph uniformly at random and the nodes connected by this edge interact. A fundamental task in this model is *stable leader election*, in which all nodes start in an identical state and the aim is to reach a configuration in which (1) exactly one node is elected as leader and (2) this node remains as the unique leader no matter what sequence of interactions follows. On *cliques*, the complexity of this problem has recently been settled: time-optimal protocols stabilize in  $\Theta(n \log n)$  expected steps using  $\Theta(\log \log n)$  states, whereas protocols that use  $O(1)$  states require  $\Theta(n^2)$  expected steps. In this work, we investigate the complexity of stable leader election on graphs. We provide the first non-trivial time lower bounds on general graphs, showing that, when moving beyond cliques, the complexity of stable leader election can range from  $O(1)$  to  $\Theta(n^3)$  expected steps. We describe a protocol that is time-optimal on many graph families, but uses polynomially-many states. In contrast, we give a near-time-optimal protocol that uses only  $O(\log^2 n)$  states that is at most a factor  $O(\log n)$  slower. Finally, we observe that for many graphs the constant-state protocol of Beauquier et al. [OPODIS 2013] is at most a factor  $O(n \log n)$  slower than the fast polynomial-state protocol, and among constant-state protocols, this protocol has near-optimal *average case complexity* on dense random graphs.

**Keywords** Leader election · Graph algorithms · Population protocols · Lower bounds · Distributed algorithms

## 1 Introduction

Leader election is one of the most fundamental symmetry-breaking problems in distributed computing [1]: given a distributed system consisting of  $n$  identical nodes, the goal is to designate exactly one node as a leader and all others as followers. In this work, we study the computational complexity of leader election in the *stochastic population protocol model*, a popular model of distributed computation among a population of (initially) indistinguishable agents that reside

on a graph and interact in an unpredictable, random manner [2, 3].

### 1.1 The stochastic population model on graphs

In the stochastic population protocol model, or simply the *population model*, the system is described by a finite, connected graph  $G = (V, E)$  with  $n$  nodes. Each node represents an agent, corresponding to a finite state automaton. Initially, all nodes are identical and anonymous.

*Model of computation* In the population model, computation proceeds asynchronously, in a series of random pairwise interactions between neighbouring nodes in the graph  $G$ . In each discrete time step,

1. the scheduler samples an ordered pair  $(u, v)$  *uniformly at random* among all pairs of nodes connected by an edge in  $G$ ,
2. the selected nodes  $u$  and  $v$  interact by exchanging information and updating their local states, and
3. every node maps their local state to an output value.

✉ Dan Alistarh  
dan.alistarh@ista.ac.at

✉ Joel Rybicki  
joel.rybicki@hu-berlin.de

✉ Sasha Voitovych  
voitovyc@mit.edu

<sup>1</sup> Institute of Science and Technology Austria, Klosterneuburg, Austria

<sup>2</sup> Humboldt University of Berlin, Berlin, Germany

<sup>3</sup> MIT, Cambridge, USA

When the scheduler selects the ordered pair  $(u, v)$  of nodes upon an interaction step, we say that  $u$  is the *initiator* of the interaction and  $v$  is the *responder*. The algorithm is described by a state transition function, which is typically given by a collection of local update rules of the form  $A + B \rightarrow C + D$ , where  $A$  and  $B$  are the states of the initiator and the responder at the start of an interaction, and  $C$  and  $D$  are the resulting states after the interaction.

*Stable leader election* In the case of leader election, nodes have two possible output values to indicate whether they are a *leader* or a *follower*. The goal is to design the local update rules so that the system reaches a *stable configuration* in which (1) exactly one node  $v \in V$  is elected as the leader and all other nodes are followers and (2) the node  $v$  remains as the unique leader no matter what sequence of interactions follows (i.e., all configurations reachable from a stable configuration have the same output).

*Complexity measures* The *time complexity* is measured by *stabilization time*, which is the *total number of interaction steps* needed to reach a stable configuration. The typical aim is to guarantee that stabilization time is small both in expectation and with high probability. Here, “with high probability” refers to success probability of at least  $1 - 1/n^\lambda$ , where  $n$  is the number of nodes in the graph and  $\lambda > 0$  can be chosen to be an arbitrary constant by adjusting constants of the underlying protocols. Finally, we measure *space complexity* as the maximum number of distinct node states employed by the protocol.

## 1.2 Prior work on leader election in the population model

Already the foundational work on population protocols raised the question how the structure of the interaction graph influences the computational power [2–4] and complexity [5] of stable computation in the population model. In particular, the complexity of stable leader election on general interaction graphs has remained an open problem. Instead, most work in this area has focused on a special case where the interaction graph is restricted to be a clique [6, 7]. While this special case naturally corresponds to *well-mixed systems*, it is often too simplistic when modelling systems where the interaction patterns among agents are influenced by some underlying spatial structure.

Leader election is known to be an especially important problem in the population model: for instance, the early work of Angluin, Aspnes and Eisenstat [8] showed that having a leader can be useful in the population model on cliques: semi-linear predicates can be stably computed in  $O(n \log^5 n)$  time, and randomized LOGSPACE computation can be performed with small error [8]. The above result has motivated a vast amount of follow-up work on the complexity of leader election on *cliques* [6, 7, 9–18]. By now, the complexity of leader

election on the clique is well-understood: there exists a protocol that solves leader election in  $\Theta(n \log n)$  expected steps using  $\Theta(\log \log n)$  states per node [18], which is optimal. To elect a leader in the clique model, all protocols require  $\Omega(n \log n)$  expected steps [17], any  $o(\log \log n)$ -state protocol requires  $n^2 / \text{polylog } n$  expected steps [11] and the optimal time complexity bound for constant-state protocols is  $\Theta(n^2)$  expected steps [9].

Somewhat surprisingly, much less is known about the complexity of leader election on general interaction graphs. Angluin, Aspnes, Fischer and Jiang [4] showed that *self-stabilizing* leader election is not generally possible on all connected interaction graphs. At the same time, Beauquier, Blanchard and Burman [19] showed that there exists a constant-state protocol that solves stable leader election as long as self-stabilization is not required. Subsequently, research on leader election in the population model has largely fallen into two categories: (1) work that tries to understand computational complexity and space-time complexity trade-offs of leader election under uniform random pairwise interactions on the clique [9, 10, 12–18, 20], and (2) work that aims to understand in which interaction graphs and under what model assumptions leader election can be solved in, e.g., a fault-tolerant manner [4, 19, 21–26].

An interesting question left open by this line of work is the *computational complexity* of stable leader election, without the requirement of self-stabilization, on *general interaction graphs* [27]. One reason is that algorithmic [6, 7, 15, 16, 18, 20] and lower bound techniques [9, 11, 17] developed for the clique model do not readily extend to the case of general interaction graphs. More broadly, establishing tight bounds for randomized leader election is known to be challenging even in well-studied synchronous models of distributed computing, such as the LOCAL and CONGEST models [28, 29].

## 1.3 Limits of existing techniques

Existing *upper bound techniques* on the clique naturally rely on the fact that every pair of nodes can potentially interact. Specifically, fast and space-efficient algorithms [15, 18, 20] combine (1) fast information dissemination dynamics, typical for the clique, with (2) careful time-keeping across “*juntas of nodes*” to obtain space-efficient phase clocks. It is not straightforward to generalize either of these techniques to, e.g., sparse or poorly-connected graphs.

The only existing work to explicitly consider the complexity of non-self-stabilizing leader election on graphs is by Alistarh, Gelashvili and Rybicki [27], whose general goal is to find general ways of porting clique-based algorithms to regular interaction graphs. (Chen and Chen [22] considered complexity of *self-stabilizing* leader election in regular graphs, but this is computationally harder than stable leader election [4].) Specifically for regular graphs,

Alistarh et al. [27] gave a leader election protocol that stabilizes in  $1/\phi^2 \cdot n \text{polylog } n$  steps in expectation and with high probability and uses  $1/\phi^2 \cdot \text{polylog } n$  states, where  $\phi$  is the conductance of the interaction graph. While this approach yields to fairly efficient leader election protocols in graphs with high conductance, it performs poorly in low-conductance graphs. For example, on cycles the protocol uses  $n^2 \text{polylog } n$  states and requires  $\Omega(n^3)$  steps to stabilize.

Alistarh et al. [27] also showed that the constant-state protocol of Beauquier et al. [19] stabilizes in the order of  $Dmn^2 \log n$  steps in expectation and with high probability on any graph with diameter  $D$  and  $m$  edges. This upper bound can be further refined to  $O(\mathbf{C}(G) \cdot n \log n)$ , where  $\mathbf{C}(G)$  is the cover time of a classic random walk on the graph  $G$ , by leveraging the recent results of Sudo, Shibata, Nakamura, Kim and Masuzawa [25]. However, beyond the case of cliques, there are no results indicating whether this bound could be improved.

Specifically, existing *lower bound techniques* for population protocols on the clique [9, 11, 17] do not directly generalize to general graphs. In particular, such approaches usually rely on the fact that short executions can lead to “populous” configurations which have large “leader generating” sets of nodes; then, by carefully interleaving interactions between nodes in such sets, short executions can be extended to create new leaders. This suggests that short executions are unlikely to yield stable configurations. However, to create new leaders, existing arguments require the set of nodes to be connected in the underlying graph. This is straightforward on the clique, but non-trivial for general graphs.

The situation seems even more challenging when trying to establish space-time complexity trade-offs, such as showing that constant-state protocols cannot run in sublinear time. In this case, the only known approach is the *surgery technique* [9, 11], which requires keeping track of the distribution of certain states that can be used to generate a leader. On *general graphs*, one would therefore also need to keep track of the *spatial distribution* of states created by the protocol, which appears highly complex for general protocols and interaction graphs.

To illustrate the difficulty of extending the above techniques to general interaction graphs, a useful exercise is to consider the case of star graphs: there is an  $O(1)$ -state protocol that elects a leader in a single interaction in any star of size  $n$ .<sup>1</sup> Thus, the lower bound for constant-state protocols or the general lower bound of  $\Omega(n \log n)$  expected steps cannot hold in general, as in some graphs, the graph structure can be used to break symmetry fast using a small number of states.

<sup>1</sup> For example, consider a protocol with states  $\{0, 1, 2\}$ , where all nodes start in a state 0, the state 2 is the unique leader state, and the only rule that can change nodes’ states is given by  $0 + 0 \rightarrow 1 + 2$ .

## 1.4 Our contributions

In this work, we give new upper bounds and lower bounds for stable leader election in the population model on general graphs. For many graph families, we obtain either tight or almost tight bounds; please see Table 1 for a summary of our results. We detail our main results in the following.

### Bounds on information propagation in the population model

We phrase our upper bounds in terms of *worst-case expected broadcast time*  $\mathbf{B}(G)$  on the graph  $G$ . Informally,  $\mathbf{B}(G)$  denotes the maximum expected time until a broadcast originating from a single node reaches all other nodes in the graph  $G$ . This process is often called “one-way epidemics” in the population protocol literature [8].

In Section 3, we establish the worst-case expected broadcast time upper bounds of  $O(mD + m \log n)$  and  $O(m/\beta \cdot \log n)$  for any  $m$ -edge graph with diameter  $D$  and edge expansion  $\beta$ . Thus,  $\mathbf{B}(G) \in O(n^3)$  for any connected graph, but it can often be much smaller. We also provide lower bounds on the time that information propagates to a given distance  $k$ . These bounds are then used to lower bound both broadcast time and leader election time for general protocols.

While for regular graphs these dynamics correspond to well-studied asynchronous rumour spreading [30, 31], when the graph is not regular, the dynamics in the population model behave differently: the interaction rate of a node depends on its degree.

**Fast space-efficient leader in close-to-broadcast time**  
We first observe that, if we disregard space complexity, there exists a simple protocol that solves leader election in  $O(\mathbf{B}(G) + n \log n)$  expected steps, on any connected graph  $G$ : nodes can generate unique identifiers, and then broadcast them to elect a leader. While this protocol is time-optimal on many graphs, generating unique identifiers will require polynomially-many states, and thus, results in a high space complexity.

Our first contribution is a *space-efficient protocol* that elects a leader in  $O(\mathbf{B}(G) \cdot \log n)$  steps in expectation and with high probability using only  $O(\log n \cdot h(G))$  states, where  $h(G) \in O(\log n)$  is a parameter depending on the broadcast time  $\mathbf{B}(G)$ . Contrasting to the identifier-based approach, this space-efficient protocol achieves exponentially smaller space complexity of  $O(\log^2 n)$ , with a factor  $O(\log n)$  increase in stabilization time.

Our protocol builds on a time-optimal approach on the clique by Sudo, Fukuhito, Izumi, Kakugawa and Masuzawa [20], and significantly improves upon the state-of-the-art on general graphs. Specifically, Alistarh, Gelashvili and Rybicki [27] gave a protocol for leader election on  $\Delta$ -regular graphs that stabilizes in  $O(n/\phi^2 \cdot \log^6 n)$  expected steps and uses  $O(\log^7 n/\phi^2)$  states per node, where  $\phi = \beta/\Delta$  is the conductance of the graph. Our protocol has stabilization time

**Table 1** Complexity bounds for stable leader election. Stabilization time refers to the expected number of steps required to reach a stable configuration. Here  $\mathbf{B}(G)$  is a characteristic of information dissemination dynamics in the population model defined in Section 3. The quantity  $\mathbf{H}(G)$  is the worst-case expected hitting time of a simple, classic random walk on  $G$ . The  $O(\log^2 n)$  and  $O(1)$ -state protocols also stabilize in the reported time with high probability. For regular graphs,

Graph family	Stabilization time	States	Reference
General	$O(\mathbf{B}(G) + n \log n)$	$O(n^4)$	Theorem 4
	$O(\mathbf{B}(G) \cdot \log n)$	$O(\log^2 n)$	Theorem 5
	$O(\mathbf{H}(G) \cdot n \log n)$	$O(1)$	[19] + [25] + Theorem 3
Renitent*	$\Omega(\mathbf{B}(G))$	$\infty$	Theorem 7
Regular	$O(\phi^{-2} n \log^6 n)$	$O(\phi^{-2} \log^7 n)$	[27]
	$O(\phi^{-1} n \log n)$	$O(n^3)$	Theorem 4
	$O(\phi^{-1} n \log^2 n)$	$O(\log n \cdot \log(\phi^{-1} \cdot \log n))$	Corollary 1
	$O(\phi^{-1} n^2 \log^2 n)$	$O(1)$	Theorem 3
Cliques	$\Theta(n \log n)$	$\Theta(\log \log n)$	[11, 17, 18]
	$\Theta(n^2)$	$O(1)$	[9]
Dense random**	$\Theta(n \log n)$	$O(n^4)$	Theorem 8 + 4
	$O(n \log^2 n)$	$O(\log^2 n)$	Theorem 5
	$o(n^2)$ impossible	$O(1)$	Theorem 9
	$O(n^2 \log^2 n)$	$O(1)$	Theorem 3
Stars	$O(1)$	$O(1)$	Trivial

$O(n/\phi \cdot \log^2 n)$  on regular graphs; this improves the dependency on the conductance  $\phi$  by a linear factor and the polylogarithmic dependence from  $\log^6 n$  to  $\log^2 n$ . In terms of space complexity, we get an *exponential* improvement in conductance, as the parameter  $h$  in the space complexity bound satisfies  $h \in O(\log \log n + \log(1/\phi))$  in regular graphs.

We emphasize that our protocol also works in *non-regular graphs*, and guarantees that the elected leader has degree  $\Theta(\Delta)$ , where  $\Delta$  is the maximum degree of the graph, with high probability. Our protocol has high-degree nodes driving a space-efficient and approximate distributed phase clock: nodes with degree  $\Theta(\Delta)$  generate “clock ticks” roughly every  $\mathbf{B}(G)$  steps with high probability. With this in place, we devise a protocol in which high-degree nodes participate in a tournament that lasts for  $O(\log n)$  phases, and each phase lasts for  $O(\mathbf{B}(G))$  steps.

### Time lower bounds for general protocols

On the negative side, we show how to construct families of graphs in which leader election and broadcast have the same asymptotic time complexity. Our approach is based on a probabilistic indistinguishability argument similar in spirit to the lower bound argument of Kutten, Pandurangan, Peleg, Robinson and Trehan [29] for randomized leader election in the synchronous LOCAL and CONGEST models. However, in the population model, communication patterns are asyn-

chronous and stochastic instead of synchronous, so we need a more refined approach to establish the lower bounds.

Roughly speaking, we show that if (a) the nodes of the graph can be divided into constantly many subsets  $V_1, \dots, V_K$  such that the local neighbourhoods of these sets are isomorphic up to some distance  $\ell$  and (b) there are sets whose distance- $\ell$  neighbourhoods are disjoint, then any leader election protocol must propagate information at least up to distance  $\ell$  to reach a stable configuration. If this propagation takes at least  $f(n)$  steps with at least a constant probability, then we get a lower bound of order  $f(n)$  for the expected stabilization time. We call such graphs  $f$ -renitent (see Section 6 for a formal definition).

In general, it is fairly straightforward to construct graphs with diameter  $\Theta(D)$  and  $\Theta(m)$  edges, which are  $\Omega(Dm)$ -renitent for any  $1 \leq D \leq n$  and  $n \leq m \leq n^2$ . Moreover, in these graphs broadcast time is  $\Theta(Dm)$ . Our proof works for a general variant of the population model, in which we do not restrict the state space of the protocol and give each node an *infinite* stream of uniform, fair random bits that assign unique identifiers for each node with probability 1 at the start of the execution. Finally, we also show that in *any* sufficiently dense graph, leader election requires  $\Omega(n \log n)$  expected steps. This part of the argument extends the lower bound argument of Sudo and Masuzawa [17] from cliques to dense graphs.

**Worst-case and average-case complexity of constant-state protocols** As a baseline result, we observe that the

constant-state protocol of Beauquier et al. [19] stabilizes in  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability, where  $\mathbf{H}(G)$  is the worst-case expected hitting time of a *classic random walk* on  $G$ . For any connected graph, it is known that  $\mathbf{H}(G) \in O(n^3)$ , and if the graph is regular, then  $\mathbf{H}(G) \in O(n^2)$  holds [32].

It follows from the analyses of Alistarh et al. [27] and Sudo et al. [25] that this protocol stabilizes in  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability. In the preliminary conference version of this paper [33], we erroneously reported a running time in terms of  $O(\mathbf{B}(G) \cdot n \log n)$ ; the analysis was unfortunately flawed. It remains an interesting question to determine if we can bound the hitting times of tokens in the population model also as a function of the broadcast time  $\mathbf{B}(G)$ , since this would immediately imply a bound for the constant-state protocol as a function of the broadcast time as well. This would allow for a cleaner comparison between the running times of the different protocols.

As our final contribution, we show that, in the class of constant-state protocols, the *average-case complexity* of the protocol by Beauquier et al. [19] on dense random graphs is optimal up to an  $O(\log n)$  factor. We show that the stabilization time of any leader election protocol, that works on all connected graphs, cannot be  $o(n^2)$  in expectation for connected Erdős–Rényi random graphs  $G \sim G_{n,p}$  where  $p > 0$  is any constant. This is tight up to a logarithmic factor, as the hitting time satisfies  $\mathbf{H}(G) \in O(n)$  asymptotically almost surely in these graphs [34]. Therefore, the 6-state protocol stabilizes in  $O(n^2 \log n)$  steps with probability  $1 - o(1)$  on a random graph  $G \sim G_{n,p}$ . As  $\mathbf{B}(G) \in \Omega(n)$  for any graph  $G$ , this implies that, on these graphs, constant-state protocol of Beauquier et al. [19] stabilizes slower than optimal by a factor of at most  $O(n \log n)$ .

To show the lower bound, we extend the *surgery technique*, used to prove space-time lower bounds for population protocols so far only in the clique [9, 11], to the case of (highly) dense random graphs.

## 2 Preliminaries

We now establish some key definitions and notation used throughout the paper.

### 2.1 Graphs

Let  $G = (V, E)$  be a undirected graph, where  $V = V(G)$  is the set of nodes and  $E(G) = E$  is the set of edges of the graph. We use  $n = |V(G)|$  to denote the number of nodes and  $m = |E(G)|$  the number of edges. The degree  $\deg(v)$  of a node  $v \in V$  is the number of edges incident to it. We use  $\Delta = \max\{\deg(v) : v \in V\}$  to denote the maximum degree and  $\delta = \min\{\deg(v) : v \in V\}$  the minimum degree of the

graph. Unless otherwise specified, we assume all graphs are connected.

Given a nonempty node set  $S \subseteq V(G)$ , the edge boundary  $\partial S$  of  $S$  is the set  $\partial S = \{\{u, v\} \in E(G) : u \in S, v \in V(G) \setminus S\}$ . The *edge expansion* of the graph  $G$  is given by

$$\beta(G) = \min \left\{ \frac{|\partial S|}{|S|} : \emptyset \neq S \subseteq V(G), |S| \leq n/2 \right\}.$$

We define  $G[S]$  to be a subgraph of  $G$  induced on vertices of  $S$ .

The distance between two vertices  $u$  and  $v$  is denoted by  $\text{dist}(u, v)$ . The radius- $r$  neighbourhood of  $u$  is  $B_r(u) = \{v \in V : \text{dist}(u, v) \leq r\}$  and

$$B_r(U) = \bigcup_{u \in U} B_r(u).$$

For  $r = 1$ , we use the short-hand  $B(u) = B_1(u)$ . The diameter  $D(G)$  is given by  $D(G) = \max\{\text{dist}(u, v) : u, v \in V(G)\}$ . For any two graphs  $G$  and  $H$ , we write  $G \simeq H$  if they are isomorphic.

For random graphs, we use the Erdős–Rényi random graph model  $G_{n,p}$ . In this model, a random graph  $G \sim G_{n,p}$  is sampled as follows. We start with  $n$  nodes and for each  $u, v \in V$  such that  $u \neq v$ , we add the edge  $\{u, v\}$  with probability  $p$  independently of all other edges.

### 2.2 Population protocols on graphs

A (stochastic) *schedule* on a graph  $G$  is an infinite sequence  $(e_t)_{t \geq 1}$  of ordered pairs of nodes  $(v, u)$ , where each  $e_t$  is sampled independently and uniformly at random among all pairs of nodes connected by an edge in  $G$  (there are  $2m$  such pairs). The order of nodes in the pair is used to distinguish between initiator and a responder. A *protocol* is a tuple  $\mathcal{A} = (\Lambda, \Xi, \Sigma_{\text{in}}, \Sigma_{\text{out}}, \text{init}, \text{out})$ , where  $\Lambda$  is the set of states,  $\Xi: \Lambda \times \Lambda \rightarrow \Lambda \times \Lambda$  is the state transition function,  $\Sigma_{\text{in}}$  and  $\Sigma_{\text{out}}$  are the sets of input and output labels, respectively,  $\text{init}: \Sigma_{\text{in}} \rightarrow \Lambda$  is the initialization function, and  $\text{out}: \Lambda \rightarrow \Sigma_{\text{out}}$  is the output function.

A configuration is a map  $x: V \rightarrow \Lambda$ , where  $x(v)$  is the state of the node  $v$  in configuration  $x$ . For any  $e = (u, v)$  and configurations  $x$  and  $x'$ , we write  $x \Rightarrow_e x'$  if  $(x'(u), x'(v)) = \Xi(x(u), x(v))$  and  $x'(w) = x(w)$  for all  $w \in V \setminus \{u, v\}$ . For any sequence  $\sigma = (e_1, \dots, e_t)$  we write  $x_0 \Rightarrow_\sigma x_t$  if  $x_i \Rightarrow_{e_{i+1}} x_{i+1}$  for each  $i \geq 0$ . We say that  $x'$  is reachable from  $x$  on  $G$  if there exists some  $k \geq 1$  and  $\sigma = (e_1, \dots, e_k)$  such that  $x \Rightarrow_\sigma x'$ . Given input  $f: V(G) \rightarrow \Sigma_{\text{in}}$ , a protocol and a schedule  $(e_t)_{t \geq 1}$ , an *execution* is the infinite sequence  $(x_t)_{t \geq 0}$  of configurations, where  $x_0 = \text{init} \circ f$  is the initial configuration and  $x_t \Rightarrow_{e_{t+1}} x_{t+1}$  for  $t \geq 0$ . Note that

throughout the paper, the time step  $t$  denotes the total number of pairwise interactions that have occurred so far.

In the case of leader election, we assume that the input is a constant function, unless otherwise specified. That is, all nodes start in the same initial state. We say that a configuration  $x$  is *correct* if  $\text{out}(x(v)) = \text{leader}$  for exactly one node  $v \in V$  and for all  $u \in V \setminus \{v\}$  we have  $\text{out}(x(u)) = \text{follower}$ . A configuration  $x$  is *stable* if for every configuration  $x'$  reachable from  $x$  we have  $\text{out}(x(v)) = \text{out}(x'(v))$  for every node  $v \in V$ . The *stabilization time* of a leader election protocol  $\mathcal{A}$  is the minimum  $t$  such that  $x_t$  is stable and correct. The *state complexity* of a protocol is  $|\Lambda|$ , the number of distinct states.

Some of the protocols we consider are non-uniform in the following sense: the state space and transition function of the protocol can depend on parameters that capture high-level structural information about the population and the interaction graph (e.g., number of nodes and edges, broadcast time or the maximum degree). However, upon initialization, all nodes receive exactly the same information. For example, nodes do not initially know their own degree or identity in the interaction graph.

### 2.3 Probability-theoretic tools

Let  $X$  and  $Y$  be real-valued random variables defined on the same probability space. We say that  $X$  *stochastically dominates*  $Y$ , written as  $Y \preceq X$ , if  $\Pr[X \geq x] \geq \Pr[Y \geq x]$  for all  $x \in \mathbb{R}$ . We start with three concentration bounds. The first is a folklore result; see, e.g., [35] for a proof. The second result gives standard Chernoff bounds for sums of Bernoulli random variables. The third result gives tail bounds on the sums of geometric random variables, via Janson [36, Theorems 2.1 and 3.1].

**Lemma 1** *Let  $X \sim \text{Poisson}(\lambda)$  be a Poisson random variable with mean  $\lambda$ . Then*

- (a)  $\Pr[X \geq c\lambda] \leq \exp(-\lambda \cdot (c-1)^2/c)$  for  $c \geq 1$ ,
- (b)  $\Pr[X \leq c\lambda] \leq \exp(-\lambda \cdot (1-c)^2/(2-c))$  for  $c \leq 1$ .

**Lemma 2** *Let  $X = Y_1 + \dots + Y_k$  be a sum of independent Bernoulli random variables with  $\Pr[Y_i = 1] = p_i$ . Then*

- (a)  $\Pr[X \geq (1+\lambda) \cdot \mathbf{E}[X]] \leq \exp(-\mathbf{E}[X] \cdot \lambda^2/(2+\lambda))$  for any  $\lambda \geq 1$ , and
- (b)  $\Pr[X \leq (1-\lambda) \cdot \mathbf{E}[X]] \leq \exp(-\mathbf{E}[X] \cdot \lambda^2/2)$  for any  $\lambda \leq 1$ .

*Note that in the special case when  $p_i = p$  for all  $1 \leq i \leq k$ , the sum  $X \sim \text{Bin}(k, p)$  is a Binomial random variable.*

**Lemma 3** *Let  $p_1, \dots, p_k \in (0, 1]$  and  $X = Y_1 + \dots + Y_k$  be a sum of independent geometric random variables with*

$Y_i \sim \text{Geom}(p_i)$ . Define  $p = \min\{p_i : 1 \leq i \leq k\}$  and  $c(\lambda) = \lambda - 1 - \ln \lambda$ . Then

- (a)  $\Pr[X \geq \lambda \cdot \mathbf{E}[X]] \leq \exp(-p \cdot \mathbf{E}[X] \cdot c(\lambda))$  for any  $\lambda \geq 1$ , and
- (b)  $\Pr[X \leq \lambda \cdot \mathbf{E}[X]] \leq \exp(-p \cdot \mathbf{E}[X] \cdot c(\lambda))$  for any  $0 < \lambda \leq 1$ .

**Lemma 4 (Wald's identity)** *Let  $(X_i)_{i \geq 1}$  be a sequence of real-valued independent and identically distributed random variables and  $N$  a non-negative integer-valued random variable independent of  $(X_i)_{i \geq 1}$ . If  $N$  and all  $X_i$  have finite expectation, then  $\mathbf{E}[X_1 + \dots + X_N] = \mathbf{E}[N] \cdot \mathbf{E}[X_1]$ .*

### 3 Bounds on information propagation

Our results will rely on notions of *broadcast time* and *propagation time* in the population model. For this, we define the following infection process on a graph  $G$ : initially, each node  $v \in V$  holds a unique message. In every step, when nodes  $u$  and  $v$  randomly interact they inform each other about *all* messages they have so far received.

The *distance- $k$  propagation time* is the minimum time until some message has reached a node at distance  $k$  from its source. The *broadcast time* is the expected time until all nodes in the network are aware of all messages. Propagation times are used in our lower bounds, whereas broadcast time appears in our upper bounds. Before we formalize these notions below, we briefly discuss some work on related, but different stochastic information propagation dynamics on graphs.

#### 3.1 Information propagation in related models

Many variants of the above broadcasting process have been studied in settings ranging from information dissemination [30, 31, 37–41] to models of epidemics [42–44]. For example, in the synchronous push-pull model [37, 40], Chierichetti, Lattanzi and Panconesi [30] first showed that broadcast succeeds with high probability in  $O(\log^4 n/\phi^6)$  rounds on graphs of conductance  $\phi$ . Subsequently, they improved the running time bound to  $O(\log n/\phi \cdot \log^2(1/\phi))$  rounds [41]. Finally, Giakkoupis [45] showed that the push-pull algorithm succeeds in  $O(\log n/\phi)$  rounds with high probability, and showed that for all  $\phi \in \Omega(1/n)$  there is a family of graphs in which this bound is tight.

In the asynchronous setting, Acan, Collevecchio, Mehranian and Wormald [38] and Giakkoupis, Nazari and Woelfel [31] studied broadcasting in the continuous-time push-pull model, where each *node* has a (probabilistic) Poisson clock that rings at unit rate. They showed that on graphs in which the protocol runs in  $T$  rounds, the asynchronous protocol runs

in  $O(T + \log n)$  continuous time. Ottino-Löffler, Scott and Strogatz [43] studied a discrete-time infection model that is similar to this asynchronous setting, and characterized broadcast time in cliques, stars, lattices and Erdős-Rényi random graphs.

Although the interaction patterns in the stochastic population model and the above asynchronous models are the same for *regular graphs*, they are different in general graphs. In the population model, instead of sampling a node and then one of its neighbours in each step, our scheduler samples an *edge*. In the continuous-time setting, this corresponds to having an independent Poisson clock at each *edge* rather than each *node* in the network. Thus, high-degree nodes interact more often than low-degree nodes in the population model.

### 3.2 Information propagation in the population model

We now define information propagation dynamics in our setting. Let  $(e_t)_{t \geq 1}$  be a stochastic schedule on a graph  $G = (V, E)$ . For each node  $v \in V$ , let  $I_0(v) = \{v\}$ . For  $t \geq 0$ , define

$$I_{t+1}(v) = \begin{cases} I_t(v) \cup I_t(u) & \text{if } e_{t+1} = (u, v) \text{ or } e_{t+1} = (v, u) \\ I_t(v) & \text{otherwise.} \end{cases}$$

Following Sudo and Masuzawa [17], we say that  $I_t(v)$  is the *set of influencers* of node  $v$  at the end of step  $t$ . Nodes in  $I_t(v)$  are nodes that can (in principle) influence the state of node  $v$  at step  $t$ . The above dynamics can be seen as a rumour spreading process, where each node starts with a unique message, and whenever two nodes interact, they inform each other about each message they hold.

**Broadcast and propagation time** Let  $T(v, u) = \min\{t : v \in I_t(u)\}$  be the minimum time until node  $u$  is influenced by node  $v$ . The *broadcast time from source  $v$*  is

$$T(v) = \max\{T(v, u) : u \in V(G)\}.$$

We define the *worst-case expected broadcast time* on  $G$  to be

$$\mathbf{B}(G) = \max\{\mathbf{E}[T(v)] : v \in V\}.$$

For each  $k \geq 0$  and  $u \in V$ , let

$$T_k(u) = \min\{T(u, v) : v \in V, \text{dist}(u, v) = k\}.$$

The *distance- $k$  propagation time* in  $G$  is  $T_k(G) = \min\{T_k(u) : u \in V\}$ . If there are no nodes at distance  $k$  from node  $u$ , then  $T_k(u) = \infty$ . Moreover,  $T_k(G) = \infty$  for all  $k > D(G)$ . Note that the expected distance- $k$  propagation time gives

lower bound for the expected broadcast time as for each  $1 \leq k \leq D(G)$  we have

$$\mathbf{E}[T_k(G)] \leq \mathbf{E}[T_D(G)] \leq \mathbf{B}(G).$$

**Sampling edge sequences** For a finite sequence  $\rho \in E^k$  of  $k$  edges, let  $X(\rho)$  be the length of the shortest prefix of the schedule  $(e_t)_{t \geq 1}$  that contains  $\rho$  as a subsequence. This corresponds to number of steps until the scheduler has sampled each edge from  $\rho$  in order (with possibly other edges in between). Note that

$$X(\rho) = Y_1 + \cdots + Y_k$$

is a sum of i.i.d. geometric random variables, where  $Y_i \sim \text{Geom}(1/m)$  is the number of steps until the  $i$ th edge of  $\rho$  is sampled after sampling the  $(i-1)$ th edge in the sequence  $\rho$ . The next lemma follows immediately from Lemma 3.

**Lemma 5** *Let  $c(\lambda) = \lambda - 1 - \ln \lambda$ . For any  $\rho \in E^k$ , we have  $\mathbf{E}[X(\rho)] = km$  and*

- (a)  $\Pr[X(\rho) > \lambda km] \leq \exp(-kc(\lambda))$  for  $\lambda \geq 1$ , and
- (b)  $\Pr[X(\rho) < \lambda km] \leq \exp(-kc(\lambda))$  for  $0 < \lambda \leq 1$ .

With the above lemma, it is fairly straightforward to establish the upper bounds on the worst-case expected broadcast time  $\mathbf{B}(G)$ . We give the details in the next section.

### 3.3 Upper-bounding the broadcast time

In this section, we give establish the following upper bound result on the broadcast time.

**Theorem 1** *Let  $G$  be a graph with  $n$  nodes,  $m$  edges, edge expansion  $\beta$  and diameter  $D$ . Then the worst-case expected broadcast time satisfies*

$$\mathbf{B}(G) \in O\left(m \cdot \min\left\{\frac{\log n}{\beta}, \log n + D\right\}\right).$$

Note that there are graphs in which  $\ln n/\beta > D$ , e.g., cycles, and  $\ln n/\beta < D$ , e.g., cliques. We will later give leader election protocols whose stabilization time is bounded as a function of  $\mathbf{B}(G)$  on any graph  $G$ . In general, for any increasing function  $T$  between  $\Omega(n \log n)$  and  $O(n^3)$ , we can find families of graphs in which both the expected broadcast time and leader election time are  $\Theta(T)$ . We give the construction in Section 6.

We first derive an upper bound of order  $m \cdot (\ln n + D)$  and then a bound of order  $m \cdot \ln n/\beta$ . For both cases, we use the following observation that certain tail bounds imply an upper bound for the expectation of the broadcast time  $T(v)$ .

**Lemma 6** Let  $n \geq 2$  and  $t^* \geq 1$ . If for each  $v \in V$  and all  $\lambda \geq 1$  the random variable  $T(v)$  satisfies

$$\Pr[T(v) > \lambda t^*] \leq \left(\frac{1}{n}\right)^\lambda,$$

then  $\mathbf{B}(G) \leq 2t^* + 1$ .

**Proof** Let  $v \in V$ . Since  $T(v)$  is a non-negative discrete random variable, we have that

$$\begin{aligned} \mathbf{E}[T(v)] &= \sum_{t=0}^{\infty} \Pr[T(v) > t] \\ &= \sum_{t=0}^{t^*} \Pr[T(v) > t] + \sum_{t=t^*+1}^{\infty} \Pr[T(v) > t] \\ &\leq t^* + 1 + \sum_{t=t^*+1}^{\infty} \Pr[T(v) > t]. \end{aligned}$$

It remains to show that the infinite sum is bounded by one. To this end, we apply the hypothesis of the lemma to each term of the infinite sum with  $\lambda(t) = t/t^*$ . Note that  $\lambda(t) \geq 1$  for  $t \geq t^*$ . We then bound the resulting sum with an integral to obtain

$$\begin{aligned} \sum_{t=t^*+1}^{\infty} \Pr[T(v) > t] &= \sum_{t=t^*+1}^{\infty} \Pr[T(v) > \lambda(t) \cdot t^*] \\ &\leq \sum_{t=t^*+1}^{\infty} \left(\frac{1}{n}\right)^{\lambda(t)} \leq \int_{t^*}^{\infty} \left(\frac{1}{n}\right)^{\lambda(t)} dt \\ &= \int_0^{\infty} \left(\frac{1}{n}\right)^{\lambda(t+t^*)} dt \\ &= \int_0^{\infty} \left(\frac{1}{n}\right)^{1+x} t^* dx = \frac{t^*}{n \ln n} \leq t^*, \end{aligned}$$

where in the last line we performed integration by substitution. Since  $v \in V$  was arbitrary, we have that  $\mathbf{E}[T(v)] \leq 2t^* + 1$  holds for all  $v \in V$ . This implies that

$$\mathbf{B}(G) = \max\{\mathbf{E}[T(v)] : v \in V\} \leq 2t^* + 1.$$

□

We now establish the first tail bound in terms of the diameter of the graph.

**Lemma 7** There exists a constant  $C$  such that for any connected graph  $G = (V, E)$  with  $n$  nodes,  $m$  edges and diameter  $D$  we have

$$\Pr[T(v) > \lambda Cm \cdot \max\{\ln n, D\}] \leq 1/n^\lambda$$

for all  $\lambda \geq 1$  and  $v \in V$ .

**Proof** Let  $u \in V$ . We first bound the probability that the propagation time  $T(v, u)$  is large. Consider a shortest path  $\rho$  of length  $k$  between  $u$  and  $v$ . Let  $X(\rho)$  be the shortest prefix of the stochastic schedule that contains the edges of  $\rho$  as a subsequence, as defined in Section 3.2. Observe that  $T(v, u) \leq X(\rho)$ .

Let  $c(\lambda) = \lambda - \ln \lambda - 1$ . Note that there exists some positive constant  $C$  such that  $c(Cx) \geq 2x$  holds for all  $x \geq 1$ . Fix smallest such constant  $C$  and define

$$\eta = \frac{\lambda C}{k} \cdot \max\{\ln n, D\}.$$

By applying Lemma 5 and the definition of  $\eta$ , we get that

$$\begin{aligned} \Pr[T(v, u) > \eta \cdot \mathbf{E}[X(\rho)]] &\leq \Pr[X(\rho) > \eta \cdot \mathbf{E}[X(\rho)]] \\ &\leq \exp(-kc(\eta)) \\ &\leq \exp(-2\lambda \ln n) \\ &= 1/n^{2\lambda}. \end{aligned}$$

By using the union bound over all  $u \in V$  and the fact that  $\lambda \geq 1$ , we get

$$\begin{aligned} \Pr[T(v) > \eta \cdot \mathbf{E}[X(\rho)]] &= \Pr[\max\{T(v, u) : u \in V\} > \eta \cdot \mathbf{E}[X(\rho)]] \\ &\leq \sum_{u \in V} \Pr[T(u, v) > \eta \cdot \mathbf{E}[X(\rho)]] \leq 1/n^\lambda. \end{aligned}$$

The claim now follows, as  $\eta \cdot \mathbf{E}[X(\rho)] = \eta km = \lambda Cm \cdot \max\{\ln n, D\}$ . □

Now Lemma 6 and Lemma 7 together immediately imply the following result.

**Lemma 8** Let  $G$  be a graph with  $n$  nodes,  $m$  edges and diameter  $D$ . Then the worst-case expected broadcast time satisfies

$$\mathbf{B}(G) \in O(m \cdot \max\{\ln n, D\}).$$

Next, we establish an upper bound of order  $m/\beta \log n$  on the broadcast time using edge expansion. We will use a similar proof strategy as above: provide a suitable tail bound and then apply Lemma 6.

**Lemma 9** Let  $G = (V, E)$  be a graph with edge expansion  $\beta > 0$ . Then there exists a constant  $C$  such that for all  $\lambda \geq 1$  and any  $v \in V$  the broadcast time  $T(v)$  satisfies

$$\Pr\left[T(v) \geq \frac{\lambda Cm \ln n}{\beta}\right] \leq \left(\frac{1}{n}\right)^\lambda.$$

**Proof** Consider a node  $v \in V$  and let  $S_t = \{u : v \in I_t(u)\}$  be the set of nodes influenced by node  $v$  at time step  $t$ .

Note that for any time step  $t \geq 0$ , the event  $|S_{t+1}| > |S_t|$  occurs if and only if the scheduler samples an edge from the boundary  $\partial S_t$  of the set  $S_t$ . By definition of edge expansion,  $|\partial S_t| \geq \beta |S_t|$  when  $|S_t| \leq n/2$  and  $|\partial S_t| \geq \beta |V \setminus S_t|$  otherwise. Since the scheduler samples edges in each step independently from all other steps, the probability of this event is at least  $\beta i/m$  when  $|S_t| = i \leq n/2$  and at least  $\beta(n-i)/m$  when  $n/2 < i \leq n-1$ . Thus, the number  $X_i$  of steps it takes from the set of influenced nodes to grow from  $i$  to  $i+1$  is stochastically dominated by the geometric random variable  $Y_i \sim \text{Geom}(p_i)$  with  $p_i = \beta i/m$  for  $1 \leq i \leq n/2$  and  $p_i = \beta(n-i)/m$  for  $n/2 < i \leq n-1$ .

Let  $Y = Y_1 + \dots + Y_{n-1}$ . By linearity of expectation, we get

$$\begin{aligned} \mathbf{E}[Y] &= \mathbf{E}\left[\sum_{i=1}^{n-1} Y_i\right] = \sum_{i=1}^{n-1} \mathbf{E}[Y_i] = \sum_{i=1}^{n-1} \frac{1}{p_i} \\ &= \frac{m}{\beta} \cdot \left( \sum_{i=1}^{\lfloor n/2 \rfloor} \frac{1}{i} + \sum_{i=\lfloor n/2 \rfloor + 1}^{n-1} \frac{1}{n-i} \right) \\ &= \frac{m}{\beta} \cdot (H_{\lfloor n/2 \rfloor} + H_{\lceil n/2 \rceil - 1}), \end{aligned}$$

where in the second line we applied the definition of  $p_i$  and  $H_k$  denotes the  $k$ th harmonic number. Since the harmonic numbers satisfy  $\ln k \leq H_k \leq 1 + \ln k$ , for all sufficiently large values of  $n$  we get that

$$\frac{m \ln n}{\beta} \leq \mathbf{E}[Y] \leq \frac{2m \ln n}{\beta}.$$

The minimum time until all nodes are influenced by  $v$  satisfies  $T(v) \leq Y$ . Therefore,

$$\begin{aligned} \Pr\left[T(v) \geq \frac{\lambda C m \ln n}{\beta}\right] &\leq \Pr\left[T(v) \geq \frac{\lambda C}{2} \cdot \mathbf{E}[Y]\right] \\ &\leq \Pr\left[Y \geq \frac{\lambda C}{2} \cdot \mathbf{E}[Y]\right]. \end{aligned}$$

Let  $c(\lambda) = \lambda - \ln \lambda - 1$  be the function from Lemma 3. We choose the constant  $C$  to be large enough so that  $c(\lambda C/2) \geq \lambda$  holds for any  $\lambda \geq 1$ . Since  $Y = Y_1 + \dots + Y_{n-1}$  is a sum of independent geometric random variables with  $Y_i \sim \text{Geom}(p_i)$  and  $p_i \geq \beta/m$ , applying the above bounds and Lemma 3 with  $p = \beta/m$  yields

$$\begin{aligned} \Pr\left[Y \geq \frac{\lambda C}{2} \cdot \mathbf{E}[Y]\right] &\leq \exp\left(-p \cdot \mathbf{E}[Y] \cdot c\left(\frac{\lambda C}{2}\right)\right) \\ &\leq \exp\left(-c\left(\frac{\lambda C}{2}\right) \ln n\right) \leq \left(\frac{1}{n}\right)^\lambda. \end{aligned}$$

□

Lemma 6 and Lemma 9 imply the following result.

**Lemma 10** *For any graph  $G$  with  $n$  nodes,  $m$  edges and edge expansion  $\beta > 0$ , the expected broadcast time satisfies*

$$\mathbf{B}(G) \in O\left(\frac{Cm \log n}{\beta}\right).$$

Now Lemma 8 and Lemma 10 together imply the bounds given in Theorem 1. Finally, we briefly remark that broadcast time on dense random graphs is  $O(n \log n)$  with high probability. This bound can be used to establish that in connected dense random graphs leader election can be solved fast (Table 1).

**Remark 1** Let  $p > 0$  be a constant and  $G \sim G_{n,p}$  with  $G$  conditioned on being connected. Then  $\mathbf{B}(G) \in O(n \log n)$  in expectation and with high probability.

**Proof sketch.** In the following, we say that an event holds with exponential probability if it happens with probability at least  $1 - \exp(-n^{\Omega(1)})$ . It is well-known that sufficiently dense random graphs have a small spectral gap. For example, from [46, Theorem 1.1] it follows that the spectral gap of a normalized Laplacian is  $o(1)$  with exponential probability for Erdős-Rényi graphs  $G_{n,p}$  with  $p \in \Theta(1)$ .

From the Cheeger inequality, it follows that conductance  $\phi$  of the graph is  $1 - o(1)$ , which implies that  $m/\beta \in n(1 + o(1))$ , since an Erdős-Rényi graph with  $p \in \Theta(1)$  is almost regular: all degrees are within  $[(1 - \varepsilon)(n - 1)p, (1 + \varepsilon)(n - 1)p]$  with exponential probability. Additionally,  $m/\beta \in O(n^3)$  deterministically for any connected graph  $G$ . Thus, Theorem 1 implies that  $\mathbf{B}(G) \in O(m/\beta \cdot \log n)$  in expectation and with high probability for  $G \sim G_{n,p}$  conditioned on connectivity.

### 3.4 Lower bounds for propagation and broadcast time

We now establish lower bounds on the propagation and broadcast time. Note that  $T(v) \geq n - 1$  is a trivial lower bound, as every node needs to interact at least once with a node that was influenced by the source node. We start with a simple bound that applies to any graph.

**Lemma 11** *For any graph  $G$  with maximum degree  $\Delta$ , we have*

$$\mathbf{B}(G) \geq m/\Delta \cdot \ln(n - 1).$$

**Proof** Let  $v \in V$ . In any given step, the probability that the number of nodes influenced by node  $v$  increases from  $i$  to  $i+1$  by one is at most  $p_{i+1} = \Delta i/m$ . Let  $X_{i+1}$  be the number of steps it takes from the set of nodes influenced by  $v$  to

grow from  $i$  to  $i + 1$ . Now  $X_{i+1}$  stochastically dominates the geometric random variable  $Y_{i+1} \sim \text{Geom}(p_{i+1})$ . Therefore,

$$\begin{aligned}\mathbf{E}[T(v)] &= \sum_{i=1}^{n-1} \mathbf{E}[X_{i+1}] \geq \sum_{i=1}^{n-1} \mathbf{E}[Y_{i+1}] \\ &= \sum_{i=1}^{n-1} \frac{m}{\Delta i} = \frac{m}{\Delta} \cdot H_{n-1},\end{aligned}$$

where the harmonic number  $H_{n-1}$  satisfies  $H_{n-1} \geq \ln(n-1)$ .  $\square$

For the lower bounds, we bound the *distance- $k$  propagation times*  $T_k(G)$  using the notion of an obstructing set, which acts as a bottleneck for information propagation. For any  $r, k \in \mathbb{N}$  such that  $r \leq k$ , we say that a set  $K \subseteq E^r$  of length- $r$  edge sequences is an  $(r, k)$ -obstructing set for node  $v \in V$  if every path  $\rho(v, u)$  from  $v$  to a node  $u$  with  $\text{dist}(u, v) = k$  contains some  $\sigma \in K$  as a subsequence. The next lemma is useful for bounding propagation times.

**Lemma 12** *Let  $K$  be a  $(r, k)$ -obstructing set for  $v$ . Then for any  $0 < \lambda \leq 1$ , we have*

$$\Pr[T_k(v) < \lambda r m] \leq |K| \cdot e^{-r c(\lambda)},$$

where  $c(\lambda) = \lambda - 1 - \ln \lambda$ .

**Proof** Let  $P$  be the set of paths of from  $v$  to any node  $u$  at distance  $k$  from  $v$ . Since any path  $\rho \in P$  contains some  $\sigma \in K$  as a subsequence, if the stochastic schedule contains  $\rho$  as a subsequence, then it also contains  $\sigma$  as a subsequence. Therefore, by using the union bound and Lemma 5, we get that

$$\begin{aligned}\Pr[T_k(v) < \lambda r m] &\leq \sum_{\rho \in P} \Pr[X(\rho) < \lambda r m] \\ &\leq \sum_{\sigma \in K} \Pr[X(\sigma) < \lambda r m] \\ &\leq |K| \cdot e^{-r c(\lambda)}.\end{aligned}$$

$\square$

We next bound  $T_k$  in terms of the maximum degree  $\Delta$  of the graph.

**Lemma 13** *If  $G$  is a graph with maximum degree  $\Delta$  and  $k \geq \ln n$ , then*

$$\Pr\left[T_k(G) < (km)/(\Delta e^3)\right] \leq 1/n.$$

**Proof** Consider some node  $v \in V$  and let  $K$  be the set of all paths of length  $k$  originating from  $v$ . If no such paths exist, then  $T_k(v) = \infty$ . Observe that  $K$  is a  $(k, k)$ -obstructing

set for  $v$  and that  $|K| \leq \Delta^k = \exp(k \ln \Delta)$ . Choose  $\lambda = 1/(e^3 \cdot \Delta)$  and let  $c(\lambda) = \lambda - 1 - \ln(\lambda)$ . Observe that

$$\begin{aligned}\ln \Delta - c(\lambda) &= \ln \Delta - (\lambda - 1 - \ln \lambda) \\ &= \ln \Delta - \lambda + 1 + \ln \lambda \\ &= \ln \Delta - \frac{1}{e^3 \Delta} + 1 - \ln(e^3 \Delta) < -2.\end{aligned}$$

Applying Lemma 12 with  $K$  and  $\lambda$  yields

$$\begin{aligned}\Pr\left[T_k(v) < \frac{km}{\Delta e^3}\right] &\leq |K| \cdot e^{-k c(\lambda)} \\ &\leq e^{k \ln \Delta} \cdot e^{-k c(\lambda)} \\ &= e^{k(\ln \Delta - c(\lambda))} \\ &\leq e^{-2k} \leq 1/n^2,\end{aligned}$$

since  $k \geq \ln n$ . The lemma follows by taking the union bound over all  $v \in V$ .  $\square$

We now can give lower bounds for the worst-case expected broadcast time in regular and bounded-degree graphs.

**Theorem 2** *Let  $G$  be a graph with  $n$  nodes and diameter  $D$ . Then the worst-case expected broadcast time  $\mathbf{B}(G)$  satisfies the following:*

- If  $G$  is regular, then  $\mathbf{B}(G) \in \Omega(n \cdot \max\{D, \log n\})$ .
- If  $G$  is a bounded-degree graph, then  $\mathbf{B}(G) \in \Theta(n \cdot \max\{D, \log n\})$ .

**Proof** We first establish the lower bounds. Note that for both graph classes Lemma 11 gives the lower bound  $\Omega(n \log n)$ . Therefore, if  $D < \ln n$ , the claimed lower bounds immediately follow. Therefore, suppose that  $D \geq \ln n$  and let  $t = (Dm)/(\Delta e^3)$ . Lemma 13 implies that  $\Pr[T_D(G) < t] \leq 1/n$ . Note that for any  $u \in V$  we have

$$T_D(G) \leq T_D(u) \leq T(u).$$

By monotonicity of expectation, we get that  $\mathbf{E}[T_D(G)] \leq \mathbf{B}(G)$ . Now

$$\begin{aligned}\mathbf{B}(G) &\geq \mathbf{E}[T_D(G)] \\ &\geq \mathbf{E}[T_D(G) \mid T_D(G) \geq t] \cdot \Pr[T_D(G) \geq t] \geq t(1 - 1/n).\end{aligned}$$

The first lower bound follows from observing that for regular graphs  $m = \Delta n/2$ . The lower bound for the second claim follows by observing that  $\Delta$  is a constant. Theorem 1 gives the upper bound for bounded-degree graphs, as  $m \in O(n)$  in such graphs and so  $\mathbf{B}(G) \in O(n \cdot (\log n + D))$ , which is of order  $n \cdot \max\{\log n, D\}$ .  $\square$

## 4 Baselines for stable leader election on graphs

In this section, we discuss two protocols, which act as our baselines for time complexity and space complexity of stable leader election on general graphs. First, we note that constant-state protocol given by Beauquier et al. [19] stabilizes in  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability, where  $\mathbf{H}(G)$  is the worst-case hitting time of a simple, *classic random walk* [32]. Second, we observe that if we allow for polynomially-many states, then there is a simple protocol that elects a leader in  $O(\mathbf{B}(G) + n \log n)$  expected steps. This protocol is time-optimal for a large class of graphs.

### 4.1 First baseline: A space-efficient protocol

Our baseline for *space-efficient protocols* is the constant-state leader election protocol given by Beauquier et al. [19]. This protocol stabilizes in any connected graph in finite expected time.

Here, we first observe the following bound on the stabilization time as a function worst-case hitting time of a *classic random walk*. This result essentially follows from the recent techniques developed by Sudo et al. [25] combined with the algorithm of Beauquier et al. [19].

**Theorem 3** *There is a 6-state protocol such that if given a nonempty set of leader candidates as input, then the protocol elects exactly one candidate as a leader in  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability, where  $\mathbf{H}(G)$  is the worst-case hitting time of a classic random walk on  $G$ .*

*Overview of the protocol* The idea of the protocol is simple: as input, we are given a nonempty set of leader candidates. That is, each node  $v$  receives a single bit  $f(v) \in \{0, 1\}$  as local input denoting whether it is a leader candidate or not. The guarantee is that if there is at least one leader candidate, then one of these will be eventually the leader.

At the start of the execution, each leader candidate creates a “black token”. In each interaction, the selected nodes swap their tokens with each other. Whenever two black tokens meet, one of them is colored white and the other token is left black. Whenever a leader candidate receives a white token, this candidate becomes a follower and removes the token from the system. Eventually exactly one black token and leader candidate remain. That is, the protocol is always correct.

The formal description of this protocol appears in [19] and [27]. In particular, [19] gives a proof of correctness, i.e., that the protocol eventually stabilizes in any connected graph. Similarly to Alistarh et al. [27], we exploit this property by using this constant-state protocol as a *backup protocol* for faster protocols that may fail to elect a unique leader with

a polynomially small probability. We note that the idea of using backup protocols was already introduced by Angluin, Aspnes and Eisenstat [8].

#### Comparison to previous analyses

The main idea for the analysis is that black and white tokens in the protocol perform random walks on the interaction graph  $G$ . Recently, Alistarh et al. [27] analysed this protocol using the meeting and hitting times of randomly walking tokens in the population model. They obtain a bound on these times through the diameter and the number of edges of the graph. Sudo et al. [25] provide a more refined analysis of the meeting and hitting times in the population model by relating them to hitting times of a classic random walk, which are well-understood for large families of graphs.

In this section, we adapt these results to obtain a upper bound on the stabilization time of the protocol of Beauquier et al. [19]. In Section 7, we show that the *average-case time complexity* of this protocol on dense random graphs is almost-optimal among constant-state protocols.

#### Classic random walks and random walks in the population model

First, we give a definition of a simple *random walk in the population model*. Let  $(e_t)_{t \geq 1}$  be the sequence of (undirected) edges sampled by the stochastic scheduler. We use  $X_t(v)$  to represent the position of a random walk started at node  $v$  at time  $t \geq 0$ . The dynamics of the random walk are given by

$$X_{t+1}(v) = \begin{cases} w & \text{if } e_{t+1} = \{X_t(v), w\} \\ X_t(v) & \text{otherwise.} \end{cases}$$

In other words, the random walk travels to the other end of an edge if the edge containing its current position is sampled.

In contrast, we use  $X_t^S(v)$  to denote the position of a *classic random walk* started at node  $v$  at time  $t \geq 0$ . For a classic random walk, we have  $X_0^S(v) = v$  and the conditional distribution  $X_{t+1}^S(v) | X_t^S(v)$  is uniform on the neighbourhood of  $X_t^S(v)$ . That is, if the classic random walk is at node  $u$ , then the next location of the random walk is sampled uniformly at random among the neighbours of  $u$ .

**Hitting times** Let  $\mathbf{H}^P(u, v)$  denote the expected time for a random walk started at node  $u$  to reach node  $v$  in the population model. Define  $\mathbf{H}^P(G) = \max\{\mathbf{H}^P(u, v) : u, v \in V(G)\}$  to be the worst-case expected hitting time in the population model. Similarly, let  $\mathbf{H}(v, u)$  denote the expected hitting time from  $u$  to  $v$  of a classic random walk, and define  $\mathbf{H}(G) = \max\{\mathbf{H}(u, v) : u, v \in V(G)\}$ . Sudo et al. [25, Lemma 2] show the following relationship between  $\mathbf{H}(G)$  and  $\mathbf{H}^P(G)$ .

**Lemma 14** *For any graph  $G$ , we have  $\mathbf{H}^P(G) \leq 27n \cdot \mathbf{H}(G)$ .*

**Meeting times** We say that a random walk *meets* another random walk at time step  $t$ , if the two random walks are

located at the opposite ends of the edge  $e_t$  sampled at time step  $t$ . Let  $\mathbf{M}(u, v)$  denote the expected time until the random walks started at nodes  $u$  and  $v$  meet.

Following Sudo et al. [25], we note that the meeting times in the population model can be bounded using the hitting times using essentially the same proof as the proof of Coppersmith, Tetali and Winkler [47, Theorem 2] for classic random walks on graphs, with minor modifications. This yields the next lemma; for the sake of completeness, we provide its proof in Appendix A.

**Lemma 15** *For any  $u \neq v$ , we have  $\mathbf{M}(u, v) \leq 2 \cdot \mathbf{H}^P(G)$ .*

With the above lemma, we can now prove the next result, which slightly refines the bounds on hitting times given by Sudo et al. [25] to also hold with high probability in addition to expectation.

**Lemma 16** *Suppose we start a random walk at every node of  $G$ . Then each random walk visits every node and meets every other random walk within  $O(\mathbf{H}(G) \cdot n \log n)$  time in expectation and with high probability.*

**Proof** Let  $v, u \in V$  and consider a random walk started at vertex  $v$ . By Lemma 14, the expected hitting time  $\mathbf{H}^P(w, u)$  is upper bounded by  $27n \cdot \mathbf{H}(G)$  for any node  $w \in V$ .

Let  $k \geq 3$  be a constant and consider  $k \log n$  time intervals of length  $54n \cdot \mathbf{H}^P(G)$ . In each such interval, the random walk started at node  $v$  hits node  $u$  with probability at least  $1/2$  by Markov's inequality, regardless of its position at the beginning of the interval. Hence, it hits node  $u$  in  $54k \cdot \mathbf{H}(G) \cdot n \log n$  steps with probability at least

$$1 - \left(\frac{1}{2}\right)^{k \log n} \geq 1 - \frac{1}{n^k}.$$

Then, by union bound over all pairs of vertices, we have for any  $k \geq 2$ ,

$$\Pr \left[ \mathbf{H}^P(G) \geq 54k \cdot \mathbf{H}(G) \cdot n \log n \right] \leq \frac{1}{n^{k-2}},$$

thus, every random walk hits every node in  $O(\mathbf{H}(G) \cdot n \log n)$  steps with high probability. To get an upper bound on the expectation, let  $T_0 = 54 \cdot \mathbf{H}(G) \cdot n \log n$  and note that for any  $n \geq 2$ ,

$$\begin{aligned} \mathbf{E} \left[ \mathbf{H}^P(G) \right] &= \sum_{\ell=0}^{\infty} \Pr \left[ \mathbf{H}^P(G) > \ell \right] \\ &\leq T_0 \cdot \sum_{k=0}^{\infty} \Pr \left[ \mathbf{H}^P(G) > k \cdot T_0 \right] \\ &\leq T_0 \cdot \left( 3 + \sum_{k=3}^{\infty} \Pr \left[ \mathbf{H}^P(G) > k \cdot T_0 \right] \right) \end{aligned}$$

$$\begin{aligned} &\leq T_0 \cdot \left( 3 + \sum_{k=3}^{\infty} \frac{1}{n^{k-2}} \right) \\ &\leq 4T_0 \in O(\mathbf{H}(G) \cdot n \log(n)) \end{aligned}$$

Now let  $v, u \in V$ , and consider random walks started at vertices  $v$  and  $u$ . By Lemma 15 and Lemma 14, the expected meeting time  $\mathbf{M}(v, u)$  is upper bounded by  $54n \cdot \mathbf{H}(G)$ . Consider  $k \log n$  intervals of length  $108n \cdot \mathbf{H}(G)$ . In each interval, the random walk started at node  $v$  meets a random walk started at node  $u$  with probability at least  $1/2$  by Markov's inequality, regardless of their positions at the start of the interval. Hence, they meet in  $108k \cdot \mathbf{H}(G) \cdot n \log n$  steps with probability at least

$$1 - \left(\frac{1}{2}\right)^{k \log n} \geq 1 - \frac{1}{n^k}.$$

Then, similarly to the above, by union bound over all pairs of vertices, every pair of random walks meet in  $O(\mathbf{H}(G) \cdot n \log n)$  time in expectation and with high probability, since  $k \geq 3$  was chosen to be an arbitrary constant.  $\square$

#### Analysis of the token-based protocol

We can finally apply the above results to analyze the token dynamics of the constant-state leader election protocol. It is easy to see that each individual black or white token follows the simple random walk in the population model defined above.

For the purposes of our analysis, when a white token disappears, we replace it by a “white ghost token”. Similarly, we treat a black token that turns white as a “black ghost token”, while keeping the white token as well. This is done so that we can reason about the meeting/hitting times without worrying about the fact that black or white token may disappear before a meeting/hitting event occurs. With the above, we are now ready to prove Theorem 3.

**Theorem 3** *There is a 6-state protocol such that if given a nonempty set of leader candidates as input, then the protocol elects exactly one candidate as a leader in  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability, where  $\mathbf{H}(G)$  is the worst-case hitting time of a classic random walk on  $G$ .*

**Proof** First, the number of steps until there remains exactly one black token in the system is  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability. This follows from Lemma 16, as all black tokens (or their ghosts) have met after  $O(\mathbf{H}(G) \cdot n \log n)$  steps (in expectation and with high probability), and every time two (non-ghost) black tokens meet, one of them turns into a white token. This implies only one black token remains in the system by this time. Indeed, if there were at least two black tokens, they would meet before that, causing one of them to turn into a white token. Moreover, the number of black tokens can never be less than one.

Suppose  $w$  white tokens remain after all but one black token have been eliminated. This means there are  $w+1$  leaders in the system. Now Lemma 16 implies that each white token (or its ghost) will hit every one of the  $w+1$  leader nodes after  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability. Therefore, there is exactly one leader candidate left in the system after  $O(\mathbf{H}(G) \cdot n \log n)$  steps in expectation and with high probability. At this point the protocol has stabilized.  $\square$

The bound in Theorem 3 simplifies, when a random Erdős-Renyi graph is considered. The result below is a direct consequence of a bound given by Löwe and Torres [34, Corollary 2.1]. The result below together with Theorem 3 gives the upper bound for constant-state protocol for dense random graphs that appears in Table 1.

**Proposition 1** *Let  $G \sim G_{n,p}$  with  $p \in \log^{\omega(1)}(n)/n$ . Then, with probability  $1 - o(1)$ , we have  $\mathbf{H}(G) \in O(n)$ .*

## 4.2 Second baseline: A time-efficient protocol

We next discuss our baseline for *fast protocols*. For this, we use a simple protocol that stabilizes in  $O(\mathbf{B}(G) + n \log n)$  expected steps using polynomially many states. The results in Section 6 show that this protocol is time-optimal for a large class of graphs, as there are graphs where leader election requires  $\Theta(\mathbf{B}(G))$  steps and  $\mathbf{B}(G) \in \Omega(n \log n)$ . The protocol is non-uniform: we assume that the nodes know the value of  $\log n$ , when initializing the protocol.

In this protocol, we first generate unique identifiers with high probability by using the stochasticity of the scheduler and a large state space. Once we have unique identifiers, we can elect the node with the largest identifier as the leader by a broadcast process. The main detail to handle is that nodes only get one random bit per interaction. Therefore, the sampling of random bits is spread over  $\Theta(\log n)$  interactions.

The only non-trivial part is to get finite expected stabilization time. This is achieved by interleaving the always-correct constant-state protocol with a broadcasting process: Once a node has generated its identifier, the node starts an instance of the constant-state protocol labelled with its own identifier and designating itself as a leader candidate in this instance. If a node encounters an instance of the constant-state protocol labeled with an identifier that is higher than the identifier of its current instance (or its own identifier), the node joins as a follower to the instance with the higher identifier. In the case that two or more nodes generated the same (highest) identifier, the constant-state protocol ensures that eventually only one leader candidate remains. Specifically, we show the next result.

**Theorem 4** *There is a protocol that uses  $O(n^4)$  states on general graphs and  $O(n^3)$  states on regular graphs that elects a leader in  $O(\mathbf{B}(G) + n \log n)$  steps in expectation.*

**Description of the time-efficient protocol** Let  $\Lambda$  denote the set of states of the constant-state protocol  $\mathcal{A}$  given by Theorem 3. Recall that the protocol can be given as input a nonempty subset of nodes out of which the leader is elected.

Let  $\text{init}: \{\text{leader}, \text{follower}\} \rightarrow \Lambda$  be the initialization function of  $\mathcal{A}$  so that  $\text{init}(\text{leader})$  gives the state in which a node starts as a leader candidate and  $\text{init}(\text{follower})$  the state in which the node has been designated as a follower. Each node  $v \in V$  maintains two local variables:

- $\text{state}(v) \in \Lambda$ , and
- $\text{id}(v) \in \{1, \dots, 2^{k+1} - 1\}$ , where  $k \in \Theta(\log n)$  is a parameter.

At the start of the protocol, each node  $v$  initializes its state variables to

$$\text{id}(v) \leftarrow 1 \quad \text{and} \quad \text{state}(v) \leftarrow \text{init}(\text{follower}).$$

During an interaction  $(v_0, v_1)$ , where  $v_0$  is the initiator and  $v_1$  is the responder, node  $v_i \in \{v_0, v_1\}$  updates its state by applying the following rules:

1. If  $\text{id}(v_i) < 2^k$ , then
  - set  $\text{id}(v_i) \leftarrow 2 \cdot \text{id}(v_i) + i$ , and
  - if now  $\text{id}(v_i) \geq 2^k$ , then initialize  $\text{state}(v_i) \leftarrow \text{init}(\text{leader})$ .
2. If  $\text{id}(v_i) < \text{id}(v_{1-i})$  and  $\text{id}(v_{1-i}) \geq 2^k$ , then set
$$\text{id}(v_i) \leftarrow \text{id}(v_{1-i}) \quad \text{and} \quad \text{state}(v_i) \leftarrow \text{init}(\text{follower}).$$
3. Update  $\text{state}(v_i)$  as per the rules of the constant-state protocol  $\mathcal{A}$ , by using  $\text{state}(v_0)$  and  $\text{state}(v_1)$  as input for the transition function of  $\mathcal{A}$ .

The three rules of the protocol are executed in sequence: first both interacting nodes execute Rule (1) in parallel, then Rule (2) in parallel, and finally Rule (3). At the end of each time step, the output of node  $v$  is defined to be the output of the constant-state protocol  $\mathcal{A}$  in state  $\text{state}(v)$ .

### Analysis of the time-efficient protocol

We say that *node  $v$  starts an instance of  $\mathcal{A}$  with the identifier  $j$*  if there exists a time step  $t > 0$  such that  $v$  executes the rule  $\text{state}(v) \leftarrow \text{init}(\text{leader})$  while  $\text{id}(v) = j$  holds. This can only happen when applying Rule (1). If node executes  $\text{state}(v) \leftarrow \text{init}(\text{follower})$  in Rule (2) at time step  $t > 0$ , then we say that *node  $v$  joins an instance of  $\mathcal{A}$  (at time step  $t$ )*.

Note that a node can join many instances, but if it joins a new instance, the new instance must have a higher identifier than the previous instance. Moreover, node can start an instance only once by Rule (1) and this must happen before

the node joins any instance. While two nodes can start an instance with the same identifier, they do so with a small probability, as shown next.

**Lemma 17** *Suppose node  $u$  starts an instance with identifier  $i$  and node  $v$  starts an instance with identifier  $j$ . Then  $\Pr[i = j] \leq 1/2^k$ .*

**Proof** We now bound the probability of the event that  $u$  and  $v$  generate the same identifier (which happens by applying the first rule  $k$  times). For node  $v$ , the probability that it is an initiator on any of its interactions is  $1/2$ . This means that the identifier of  $v$  will be uniformly distributed across the set  $\{2^k, \dots, 2^{k+1} - 1\}$ . While the identifiers are *not* in general independent, we show that the identifiers will be the same with probability at most  $1/2^k$ . To this end, we distinguish three cases:

1. If  $u$  and  $v$  assign their  $b$ th bits at the same time step  $t > 0$ , then one of them will be the initiator and one the responder. Therefore, the  $b$ th bit and the identifiers of  $u$  and  $v$  will be different with probability 1.
2. If  $u$  and  $v$  never interact before assigning all of their  $k$  bits in their identifiers, then their identifiers are different with probability  $1/2^k$ . In particular, the  $b$ th bits will be different with probability  $1/2$ .
3. If  $v$  assigns its  $b$ th bit at an earlier time step  $t > 0$  than node  $u$ , then  $v$  assigns the bit at position  $b$  to 0 or 1 with equal probability, regardless of who is its interaction partner or previous bit assignments of  $v$  and  $u$ . Thus, the probability that  $u$  and  $v$  have the same bit in position  $b$  is at most  $1/2$ .

From the above, we have that nodes  $u$  and  $v$  have the same  $b$ th bit, for  $1 < b \leq k + 1$ , in their generated identifiers with probability at most  $1/2$ . Therefore, the probability that all of the  $k$  assigned bits are equal is at most  $1/2^k$ .  $\square$

Let  $\text{id}_t(v)$  be the value of the state variable  $\text{id}(v)$  after  $t$  interactions. Define  $M(t) = \max\{\text{id}_t(v) : v \in V\}$  to be the maximum  $\text{id}$  value that any node has at time step  $t$  and  $M = \max\{M(t) : t \geq 0\}$  the maximum value attained during the entire execution. Note that  $M$  is always bounded. Let  $T$  be the minimum time until all nodes have either started an instance of the constant-state protocol  $\mathcal{A}$  with identifier  $M$  or joined an instance with identifier  $M$ . (We show below that  $T$  is in fact finite.)

Note that by time  $T$  all nodes start executing the same instance of the constant-state protocol  $\mathcal{A}$ , which is guaranteed to stabilize in finite expected time. In particular, after time  $T$  if there are more than one leader, then  $\mathcal{A}$  will reduce the number of leaders to one. From Lemma 17 it follows that the probability that there is more than one leader is small.

**Lemma 18** *We have  $\mathbf{E}[T] \leq kn + 2 \cdot \mathbf{B}(G)$ .*

**Proof** Let  $v \in V$  be a node with  $\deg(v) = \Delta$  and  $X$  be the number of steps until node  $v$  has been activated  $k$  times. Clearly, node  $v$  satisfies  $\text{id}_X(v) \geq 2^k$  at time step  $X$ , as node  $v$  has either executed Rule (1) of the algorithm  $k$  times or it has satisfied the condition in Rule (2) by this time. Let  $T(v)$  denote the number of steps until a broadcast initiated at node  $v$  at time step  $X$  reaches all nodes. Therefore, for all  $t \geq X + T(v)$ , we have that  $\text{id}_t(u) \geq 2^k$  for all  $u \in V$  and no node will satisfy the condition in Step (1) at time step  $t$ .

Note that  $M(t) = M(t')$  for all  $t' \geq t \geq X + T(v)$ . Now let  $u$  be the node that attained the maximum  $\text{id}$  value at time step  $X + T(v)$ . The broadcast initiated from node  $u$  at time step  $X + T(v)$  will reach all nodes in the system by some random time  $T(u)$ . Hence, by time step  $X + T(v) + T(u)$  all nodes have the same  $\text{id}$  value. Hence,  $T \leq X + T(v) + T(u)$ . By monotonicity and linearity of expectation,

$$\mathbf{E}[T] \leq \mathbf{E}[X] + \mathbf{E}[T(v)] + \mathbf{E}[T(u)].$$

Since  $X = Y_1 + \dots + Y_k$  is the sum of  $k$  geometric random variables with mean  $m/\deg(v) = m/\Delta \leq n$ , we get that  $\mathbf{E}[X] \leq kn$ . As  $\mathbf{E}[T(v)] + \mathbf{E}[T(v)] \leq 2\mathbf{B}(G)$  by definition of the broadcast time, we get that  $\mathbf{E}[T] \leq kn + 2\mathbf{B}(G)$ .  $\square$

**Theorem 4** *There is a protocol that uses  $O(n^4)$  states on general graphs and  $O(n^3)$  states on regular graphs that elects a leader in  $O(\mathbf{B}(G) + n \log n)$  steps in expectation.*

**Proof** Let  $F$  be the event that more than one node starts an instance with identifier  $M$  and let  $v$  be any such node. The probability that some other node  $u$  starts an instance with identifier is at most  $\Pr[F] \leq n/2^k$  by using Lemma 17 and the union bound over all nodes. Let  $T'$  be the stabilization time of protocol  $\mathcal{A}$ . Using Lemma 18, the expected stabilization time of the overall protocol is

$$\begin{aligned} \mathbf{E}[T \mid \overline{F}] \cdot \Pr[\overline{F}] + \mathbf{E}[T' \mid \overline{F}] \cdot \Pr[F] \\ \leq \mathbf{E}[T] + \mathbf{E}[T'] \cdot \Pr[F] \\ \leq kn + 2 \cdot \mathbf{B}(G) + \mathbf{E}[T'] \cdot n/2^k. \end{aligned}$$

Because the worst-case hitting time of a classic random walk on general connected graphs is  $O(n^3)$  and  $O(n^2)$  on regular graphs [32], by Theorem 3, we have  $\mathbf{E}[T'] \leq Cn^4 \log n$  for some constant  $C$  for any connected graph and  $\mathbf{E}[T'] \leq Cn^3 \log n$  for any connected, regular graph. For general graphs, we can set the parameter controlling the identifier space to  $k = \lceil 4 \log n \rceil$  so that number of states is  $O(n^4)$  and the expected stabilization time becomes  $O(\mathbf{B}(G) + n \log n)$ . For regular graphs, it suffices to set  $k = \lceil 3 \log n \rceil$  to get the desired bound.  $\square$

## 5 Space-efficient and fast leader election

We now give a leader election protocol whose stabilization time is parameterized by the worst-case expected broadcast time  $\mathbf{B}(G)$  and whose state complexity depends on the expansion properties of the graph. The protocol uses tournament style leader election, an approach in many leader election protocols in the clique [10, 12, 15, 20]. The idea is to prune the number of leaders iteratively so that in each iteration currently remaining leader candidates flip a coin, and if some candidate got 1 as their coin flip, all candidates who got 0 get removed.

In particular, our approach is inspired by a time-optimal algorithm on the clique due to Sudo et al. [20], but with significant differences: for instance, our algorithm works on any connected graph and guarantees that a high-degree node is elected as a leader. The main challenge is that when moving from cliques to general graphs, nodes can interact at different rates (depending on their degree) and time to disseminate information in the graph can depend on the structure of the graph.

**Theorem 5** *For any connected graph  $G$  with maximum degree  $\Delta$ , there is a leader election protocol that uses  $O(\log n \cdot h(G))$  states and stabilizes in  $O(\mathbf{B}(G) \cdot \log n)$  steps in expectation and with high probability, where  $h(G) \in O(\log(\Delta/\beta \cdot \log n))$  and  $\beta$  is the edge expansion of  $G$ .*

The protocol is non-uniform: when initializing the protocol, we assume that the nodes know the values (or upper bounds up to additive constants) of  $\log(\mathbf{B}(G) \cdot \Delta/m)$  and  $\log n$ . Observe that parameter  $h(G) \in O(\log n)$ , so the protocol uses  $O(\log^2 n)$  states. Moreover, for graphs where the ratio  $\Delta/\beta$  is small, we can obtain  $o(\log^2 n)$  space complexity. For example, in regular graphs we get the following bounds.

**Corollary 1** *In any regular graph with conductance  $\phi = \beta/\Delta$ , there is a leader election protocol that stabilizes in  $O(1/\phi \cdot n \log^2 n)$  steps in expectation and with high probability using  $O(\log n \cdot (\log \log n - \log \phi))$  states.*

The algorithm consists of three parts. First, we describe a space-efficient way for nodes to approximately count the number of *local* interactions. The second part is a two-phase protocol that first removes low-degree nodes from the set of leader candidates and then reduces the number of high-degree leader candidates to one with high probability. Finally, to guarantee finite expected stabilization time, we use the constant-state token-based leader election protocol given in Theorem 3 as a backup protocol to handle the unlikely cases where the fast part fails.

### 5.1 Local approximate clocks on graphs

We first describe a subroutine that is used to trigger events every  $\Theta(2^h)$  expected interactions using exactly  $h + 1$  local states, where  $h \geq 1$  is a given parameter controlling the frequency of the triggered events. Each node  $v$  maintains a variable  $\text{streak}(v) \in \{0, \dots, h\}$ , which is initialized to 0. When node  $v$  interacts, it updates its streak counter as follows:

- If  $v$  is the initiator, then set

$$\text{streak}(v) \leftarrow \text{streak}(v) + 1.$$

Otherwise, set

$$\text{streak}(v) \leftarrow 0.$$

- If  $\text{streak}(v) = h$ , then node  $v$  is said to *complete a streak* and set

$$\text{streak}(v) \leftarrow 0.$$

In the above clock protocol, a local event is triggered at a node whenever the node completes a streak.

#### Analysis of the clock protocol

Let  $K$  denote the number of times a fixed node needs to interact until it first completes a streak. Here  $K$  is the number of fair coin flips needed to observe  $h$  consecutive heads, as the scheduler picks the role of initiator and responder uniformly at random and independently from previous interactions.

We start with a technical result that approximates the distribution of  $K$  using geometric random variables. Recall that  $X \preceq Y$  denotes that  $Y$  stochastically dominates  $X$ .

**Lemma 19** *The random variable  $K$  satisfies  $Z_0 \preceq K \preceq Z_1 + h$ , where  $Z_0 \sim \text{Geom}(2^{-h})$  and  $Z_1 \sim \text{Geom}(2^{-h-1})$ .*

The proof of Lemma 19 is given in Appendix B. We use  $X(d)$  to denote the number of *steps* (i.e., the number of node pairs sampled by the scheduler) until a fixed node of degree  $d$  completes a streak. Note that high degree nodes have a higher probability to complete their streaks, as they interact more often.

The next lemma summarizes some useful properties of  $K$  and  $X(d)$ . The proof follows by application of concentration bounds on geometric random variables (Lemma 3) and Wald's identity (Lemma 4).

**Lemma 20** *Let  $1 \leq d \leq n$ . The random variables  $K$  and  $X(d)$  satisfy the following:*

- The expected value of  $K$  is  $\mathbf{E}[K] = 2^{h+1} - 2$ .*
- The expected value of  $X(d)$  is  $\mathbf{E}[X(d)] = \mathbf{E}[K] \cdot m/d$ .*

(c) For any  $0 \leq \lambda \leq 1$  we have  $\Pr[X(d) \leq \lambda \cdot \mathbf{E}[X(d)]] \leq 4\lambda + 2^{1-\lambda h}$ .

**Proof** For claim (a), let  $s(h)$  denote the expected number of fair coin flips to obtain  $h$  consecutive heads. By law of total expectation,  $s(h)$  satisfies the recurrence

$$s(h+1) = \frac{1}{2}(s(h) + 1) + \frac{1}{2}(s(h) + s(h+1) + 1)$$

Rewriting this yields  $s(h+1) = 2s(h) + 2$ . Noting that  $s(1) = 2$  and solving this recurrence yields  $\mathbf{E}[K] = s(h) = 2^{h+1} - 2$ .

For claim (b) and claim (c), note that  $X(d)$  is the random sum

$$X(d) = Y_1 + \cdots + Y_K$$

of  $K$  independent and identically distributed geometric random variables, where  $Y_i \sim \text{Geom}(d/m)$  is the number of steps until the fixed node of degree  $d$  interacts for the  $i$ th time after its  $(i-1)$ th interaction for all  $i \geq 1$ . Since the random variable  $K$  is independent of the sequence  $(Y_i)_{i \geq 1}$ , Wald's identity (Lemma 4) implies

$$\mathbf{E}[X(d)] = \mathbf{E}[K] \cdot \mathbf{E}[Y_1] = \mathbf{E}[K] \cdot m/d,$$

which establishes claim (b).

Finally, to prove claim (c), set  $k = \lceil 2\lambda \cdot \mathbf{E}[K] \rceil$  and define  $X = Y'_1 + \cdots + Y'_k$ , where  $Y'_i \sim \text{Geom}(d/m)$  and all  $Y'_i$ 's are independent. By linearity of expectation, we have

$$\mathbf{E}[X] = \frac{km}{d} = \frac{k}{\mathbf{E}[K]} \cdot \mathbf{E}[X(d)] \geq 2\lambda \cdot \mathbf{E}[X(d)].$$

Then conditioned on the event  $K \geq k$ , the random variable  $X(d)$  stochastically dominates  $X$ . That is,  $\Pr[X(d) \geq x \mid K \geq k] \geq \Pr[X \geq x]$ . This means that

$$\begin{aligned} \Pr[X(d) \leq \lambda \cdot \mathbf{E}[X(d)] \mid K \geq k] &\leq \Pr[X \leq \lambda \cdot \mathbf{E}[X(d)]] \\ &\leq \Pr[X \leq \mathbf{E}[X]/2] \\ &\leq \exp\left(-\frac{d}{m} \cdot \mathbf{E}[X] \cdot c(1/2)\right) \leq \exp(-k/4) \\ &\leq \exp(-\lambda \cdot 2^{h-1}) \leq 2^{-\lambda h}, \end{aligned}$$

where the third inequality follows from Lemma 3b and last inequality follows from  $2^{h-1} \geq h$ . Using the law of total probability, we get

$$\begin{aligned} \Pr[X(d) \leq \lambda \cdot \mathbf{E}[X(d)]] &= \Pr[X(d) \geq \lambda \cdot \mathbf{E}[X(d)] \mid K < k] \\ &\quad \cdot \Pr[K < k] \end{aligned}$$

$$\begin{aligned} &+ \Pr[X(d) \geq \lambda \cdot \mathbf{E}[X(d)] \mid K \\ &\geq k] \cdot \Pr[K \geq k] \\ &\leq \Pr[K < k] + \Pr[X(d) \leq \lambda \\ &\cdot \mathbf{E}[X(d)] \mid K \geq k] \\ &\leq \Pr[K < k] + 2^{-\lambda h}. \end{aligned}$$

Recall that for a geometric random variable  $Z \sim \text{Geom}(1/2^h)$ , we have

$$\Pr[Z \leq k] = 1 - (1 - 1/2^h)^k.$$

Note also that  $k \leq 2\lambda \cdot \mathbf{E}[K] + 1 \leq \lambda 2^{h+2} + 1$ . The claim now follows using Lemma 19 and applying the Bernoulli inequality to the term  $\Pr[K < k]$ , as

$$\begin{aligned} \Pr[K < k] &\leq \Pr[Z < k] \leq 1 - (1 - 1/2^h)^k \leq \frac{k}{2^h} \\ &\leq 4\lambda + 2^{-h}. \end{aligned}$$

Therefore, since  $\lambda \leq 1$ , it follows that

$$\Pr[X(d) \leq \lambda \cdot \mathbf{E}[X(d)]] \leq 4\lambda + 2^{-h} + 2^{-\lambda h} \leq 4\lambda + 2^{1-\lambda h}.$$

□

Together with the above results, we can show that the number  $R$  of interactions to complete  $\ell \geq \ln n$  streaks is strongly concentrated around the interval  $[\mathbf{E}[R]/2, 4 \cdot \mathbf{E}[R]]$ . Thus, the above process can be essentially used as space-efficient local clock which ticks at a desired (approximate) frequency. In the following, we write  $c(\lambda) = \lambda - 1 - \ln \lambda$  for any  $\lambda > 0$ .

**Lemma 21** *Let  $\ell \geq \ln n$  and  $R$  be the number of interactions a node needs to complete  $\ell$  streaks. Then*

- (a) *The expected value of  $R$  is  $\mathbf{E}[R] = (2^{h+1} - 2)\ell$ .*
- (b)  *$\Pr[R \leq \lambda \cdot \mathbf{E}[R]/2] \leq 1/n^{c(\lambda)}$  for all  $0 < \lambda \leq 1$*
- (c)  *$\Pr[R \geq 4\lambda \cdot \mathbf{E}[R]] \leq 1/n^{c(\lambda)}$  for all  $\lambda \geq 1$*

**Proof** Note that (a) follows immediately from linearity of expectation and the fact that  $R$  is a sum of  $\ell$  i.i.d. copies of  $K$ . To show (b) and (c), define the following sums

$$R_0 = Y_1 + \cdots + Y_\ell \quad \text{and} \quad R_1 = Y'_1 + \cdots + Y'_\ell$$

of independent geometric random variables, where  $Y_i \sim \text{Geom}(1/2^h)$  and  $Y'_i \sim \text{Geom}(1/2^{h+1})$  for each  $1 \leq i \leq \ell$ . By linearity of expectation, we have that

$$\mathbf{E}[R_0] = \ell 2^h \quad \text{and} \quad \mathbf{E}[R_1] = \ell 2^{h+1}.$$

Lemma 19 implies that  $R_0 \preceq R \preceq R_1 + \ell h$ . Note that  $R_0 \preceq R$  implies

$$\begin{aligned} \Pr[R \leq \lambda \cdot \mathbf{E}[R]/2] &= \Pr[R \leq \lambda \cdot \ell(2^h - 1)] \\ &\leq \Pr[R \leq \lambda \ell 2^h] \\ &= \Pr[R \leq \lambda \mathbf{E}[R_0]] \\ &\leq \Pr[R_0 \leq \lambda \mathbf{E}[R_0]] \\ &\leq e^{-\ell c(\lambda)} \leq n^{-c(\lambda)}, \end{aligned}$$

where the last two inequalities follow from Lemma 3 and  $\ell \geq \ln n$ . Similarly,  $R \preceq R_1 + \ell h$  implies

$$\begin{aligned} \Pr[R \geq 4\lambda \cdot \mathbf{E}[R]] &= \Pr[R \geq \lambda \ell(2^{h+3} - 8)] \\ &\leq \Pr[R \geq \lambda \ell 2^{h+2}] \\ &= \Pr[R \geq 2\lambda \cdot \mathbf{E}[R_1]] \\ &\leq \Pr[R_1 + \ell h \geq 2\lambda \ell 2^{h+1}] \\ &= \Pr[R_1 \geq 2\lambda \ell 2^{h+1} - \ell h] \\ &\leq \Pr[R_1 \geq \lambda \ell 2^{h+1}] \\ &\leq e^{-\ell c(\lambda)} \leq n^{-c(\lambda)}, \end{aligned}$$

where the third inequality follows from  $\lambda \geq 1$  and  $\ell h \leq \ell 2^{h+1}$  and the last two from Lemma 3 and  $\ell \geq \ln n$ .  $\square$

Finally, we examine the concentration of the number of steps until a node completes a certain number of streaks.

**Lemma 22** Suppose  $h \in \omega(1)$ . Let  $\ell \geq \ln n$  and  $S = S(d, \ell)$  be the number of steps until a fixed node of degree  $d$  completes  $\ell$  streaks. Then for all sufficiently large  $n$ ,

- (a)  $\mathbf{E}[S] = \mathbf{E}[K] \cdot \ell m/d = (2^{h+1} - 2) \cdot \ell m/d$ ,
- (b)  $\Pr[S \leq \lambda^2 \cdot \mathbf{E}[S]/4] \leq 2/n^{c(\lambda)}$  for any  $\lambda \leq 1$ , and
- (c)  $\Pr[S \geq 8\lambda^2 \cdot \mathbf{E}[S]] \leq 2/n^{c(\lambda)}$  for any  $\lambda \geq 1$ .

**Proof** Let  $(Y_i)_{i \geq 1}$  an infinite sequence of i.i.d. geometric random variables, where  $Y_i \sim \text{Geom}(d/m)$ . Note that  $S$  is the random sum  $S = Y_1 + \dots + Y_R$ , where  $R$  is the (random) number of interactions node takes to complete  $\ell$  streaks. Since  $R$  is independent of the sequence  $(Y_i)_{i \geq 1}$ , Wald's inequality (Lemma 4) implies that  $\mathbf{E}[S] = \mathbf{E}[R] \cdot \mathbf{E}[Y_1]$ . Together with Lemma 21a this implies claim (a) of the lemma, as

$$\mathbf{E}[S] = \mathbf{E}[R] \cdot \mathbf{E}[Y_1] = \left(2^{h+1} - 2\right) \cdot \left(\frac{\ell m}{d}\right).$$

To show claim (b), let  $\lambda \leq 1$  and define

$$r_0 = \lfloor \lambda \cdot \mathbf{E}[R]/2 \rfloor \quad \text{and} \quad S_0 = \sum_{i=1}^{r_0} Y_i.$$

Note that since  $\ell \geq \ln n$ , for large enough  $n$  we have

$$\lambda \cdot \mathbf{E}[R]/4 \leq r_0 \leq \lambda \cdot \mathbf{E}[R]/2.$$

Observe that  $\mathbf{E}[S_0] \geq \lambda \mathbf{E}[S]/4$ . If the event  $R \geq r_0$  happens, then  $S_0 \leq S$ . Hence,

$$\begin{aligned} \Pr[S \leq \lambda^2 \cdot \mathbf{E}[S]/4 \mid R \geq r_0] &\leq \Pr[S_0 \leq \lambda^2 \cdot \mathbf{E}[S]/4] \\ &\leq \Pr[S_0 \leq \lambda \cdot \mathbf{E}[S_0]] \\ &\leq e^{-r_0 \cdot c(\lambda)} \leq 1/n^{c(\lambda)}, \end{aligned}$$

where the last two inequalities follow from Lemma 3 and the fact that for large enough  $n$  we have  $r_0 \in \Omega(\ell 2^h) \subseteq \omega(\log n)$ . By law of total probability,

$$\begin{aligned} \Pr[S \leq \lambda^2 \cdot \mathbf{E}[S]/4] &= \Pr[S \leq \lambda^2 \cdot \mathbf{E}[S]/4 \mid R \geq r_0] \cdot \Pr[R \geq r_0] \\ &\quad + \Pr[S \leq \lambda^2 \cdot \mathbf{E}[S]/4 \mid R < r_0] \cdot \Pr[R < r_0] \\ &\leq \Pr[S \leq \lambda^2 \cdot \mathbf{E}[S]/4 \mid R \geq r_0] + \Pr[R < r_0] \\ &\leq 1/n^{c(\lambda)} + 1/n^{c(\lambda)}, \end{aligned}$$

as the term  $\Pr[R < r_0]$  is at most  $1/n^{c(\lambda)}$  by Lemma 21.

Finally, the proof of the claim (c) follows a similar pattern. Let  $\lambda \geq 1$  and define

$$r_1 = \lceil \lambda \cdot 4 \mathbf{E}[R] \rceil \quad \text{and} \quad S_1 = \sum_{i=1}^{r_1} Y_i.$$

Note that since  $\ell \geq \ln n$ , for large enough  $n$  we have

$$\lambda \cdot 8 \mathbf{E}[R] \geq r_0 \geq \lambda \cdot 4 \mathbf{E}[R].$$

Observe that  $8\lambda \cdot \mathbf{E}[S] \geq \mathbf{E}[S_1]$ . If  $R < r_1$ , then  $S < S_1$ . Now we get that

$$\begin{aligned} \Pr[S \geq 8\lambda^2 \cdot \mathbf{E}[S] \mid R < r_1] &\leq \Pr[S_1 \geq 8\lambda^2 \cdot \mathbf{E}[S]] \\ &\leq \Pr[S_1 \geq \lambda \cdot \mathbf{E}[S_1]] \\ &\leq e^{-r_1 \cdot c(\lambda)} \leq 1/n^{c(\lambda)}, \end{aligned}$$

where the last two inequalities follow from Lemma 3 and the fact that  $r_1 \in \Omega(\ell 2^h) \subseteq \omega(\log n)$ . By law of total probability,

$$\begin{aligned} \Pr[S \geq 8\lambda^2 \cdot \mathbf{E}[S]] &= \Pr[S \geq 8\lambda^2 \cdot \mathbf{E}[S] \mid R \geq r_1] \cdot \Pr[R \geq r_1] \\ &\quad + \Pr[S \geq 8\lambda^2 \cdot \mathbf{E}[S] \mid R < r_1] \cdot \Pr[R < r_1] \\ &\leq \Pr[R \geq r_1] + \Pr[S \geq \lambda^2 \cdot \mathbf{E}[S] \mid R < r_1] \\ &\leq 1/n^{c(\lambda)} + 1/n^{c(\lambda)}, \end{aligned}$$

as  $\Pr[R \geq r_1] \geq 1/n^{c(\lambda)}$  by Lemma 21 and the second term we bounded above.  $\square$

## 5.2 The fast leader election protocol

With the time-keeping mechanism in place, we now describe and analyse the leader election protocol that reduces the number of leader candidates to *one*, with high probability, in  $O(\mathbf{B}(G) \cdot \log n)$  steps.

Let  $\tau \geq 1$  be an arbitrary fixed constant that controls the probability that the protocol fails (increasing  $\tau$  decreases the probability of failure). Fix the parameters

$$h = 8 + \left\lceil \log \left( \frac{\mathbf{B}(G) \cdot \Delta}{m} \right) \right\rceil \quad \text{and} \quad L = \lceil 2\tau \log n \rceil,$$

where  $\Delta$  denotes the maximum degree of the graph  $G$ .

Note that with this choice of parameters  $X(d) \in O(\mathbf{B}(G))$  in expectation and with high probability for any  $d \leq \Delta$  and also  $X(d) \in \Theta(\mathbf{B}(G))$  in expectation and with high probability for  $d \in \Theta(\Delta)$ . We also note that  $h \in \Omega(\log \log n)$ , as in any graph  $\mathbf{B}(G) \geq m \ln(n-1)/\Delta$ ; see Lemma 11.

**The fast, space-efficient leader election protocol** As a subroutine, each node runs the streak counter protocol with  $h$  fixed as above. In addition, every node  $v$  maintains two state variables  $\text{status}(v) \in \{\text{leader}, \text{follower}\}$  and a counter  $\text{level}(v) \in \{0, \dots, \alpha(\tau) \cdot L\}$ , where  $\alpha(\tau) > 1$  is a constant we fix later in the analysis.

Initially, each node  $v$  initializes the variables to  $\text{status}(v) \leftarrow \text{leader}$  and  $\text{level}(v) \leftarrow 0$ . When  $v$  interacts with  $u$ , node  $v$  updates its state using the following rules applied in sequence:

1. If  $v$  completes a streak and  $\text{status}(v) = \text{leader}$ , then set

$$\text{level}(v) \leftarrow \min\{\text{level}(v) + 1, \alpha(\tau) \cdot L\}.$$

2. If  $\text{level}(v) < \text{level}(u)$  and  $\text{level}(u) \geq L$ , then set

$$\text{status}(v) \leftarrow \text{follower}.$$

3. If  $\max\{\text{level}(u), \text{level}(v)\} \geq L$ , then set

$$\text{level}(v) = \max\{\text{level}(u), \text{level}(v)\}.$$

**Analysis of the fast, space-efficient leader election protocol** We now analyse the above protocol. We say that a node is at level  $\ell$  at time step  $t$  if its  $\text{level}$  variable is  $\ell$  at time step  $t$ . A node is in the *elimination phase* if it is at least at level  $L$ . Otherwise, it is in the *waiting phase*. When a node  $v$  in the waiting phase interacts with a node in the elimination phase, then  $v$  moves to the elimination phase (as a follower).

Note that a node can only remain a leader and increase its level if it completes a streak. When a node increases its level without completing a streak, it must become a follower by

Rules (2) and (3). Moreover, by Rule (2), in every time step one of the nodes in the graph with the highest level must be a leader. Thus, the protocol guarantees that there is always at least one leader in every step. Finally, as  $\mathbf{B}(G)$  is the worst-case expected broadcast time, Rule (3) implies that if some node is at level  $\ell \geq L$  at step given step, then within  $\mathbf{B}(G)$  expected steps all nodes are at level at least  $\ell$ .

The first step in the analysis considers fixed pairs of nodes, characterizing the period of time after which at least one node from a given pair drops out of contention.

**Lemma 23** *Let  $u$  and  $v$  be nodes with degree at least  $d$ . If  $u$  and  $v$  have level at least  $L$  and less than  $\alpha(\tau) \cdot L$  at step  $t$ , then at least one of them is a follower at time step  $t + 16 \cdot (\mathbf{E}[X(d)] + \mathbf{B}(G))$  with probability at least 1/8.*

**Proof** Let  $L \leq \ell < \alpha(\tau) \cdot L$  be the maximum of the levels of  $u$  and  $v$  at step  $t$ . Let  $X$  be the minimum number of steps until the one of the nodes increments its level to  $\ell + 1$  after step  $t$ . Since both nodes have degree at least  $d$ , we get that  $X \preceq X(d)$ , which implies

$$\Pr[X \geq 16 \cdot \mathbf{E}[X(d)]] \leq \Pr[X(d) \geq \mathbf{E}[X(d)]] \leq 1/16$$

by Markov's inequality. If either  $u$  or  $v$  is not leader at step  $t + X$ , then the claim follows. Suppose that both are leader candidates at step  $t + X$ . This means that only one of the nodes interacts at step  $t + X$ : if both interact at step  $t + X$ , then the initiator will complete its streak, increasing its level to  $\ell + 1$ , and the responder will fail to complete its streak and become a follower due to Rule (2).

Without loss of generality, suppose that node  $v$  interacts at step  $t + X$ . Denote  $T$  to be the number of steps it takes for broadcast from  $v$  to reach  $u$ . Then  $T \sim T(v, u)$ . Note that if by the time  $t + X + T$  node  $u$  has not completed a streak, then it is at level at most  $\ell$  and will be eliminated by the broadcast from  $v$  by Rule (2). Let  $Y$  be the number of steps after time  $t + X$  until  $u$  completes a streak.

Note that the probability that  $u$  is a leader at time  $t + X + 16 \cdot \mathbf{B}(G)$  is at most  $\Pr[T > 16 \cdot \mathbf{B}(G)] + \Pr[Y \leq 16 \cdot \mathbf{B}(G)]$ . We show that this is at most 3/4. The bound on the first term follows from Markov's inequality and the fact that  $\mathbf{B}(G)$  is the worst-case expected broadcast time, as

$$\Pr[T \geq 16 \cdot \mathbf{B}(G)] \leq \Pr[T \geq 16 \cdot \mathbf{E}[T]] \leq 1/16.$$

To bound the second term, let  $A$  denote the event that during its first interaction after time  $t + X$ , node  $u$  resets its streak counter (i.e.,  $u$  is a responder). Clearly,  $\Pr[A] = 1/2$ . Note that  $Y \mid A \preceq X(d) \preceq X(\Delta)$ , and in particular by Lemma 20b we have

$$\mathbf{E}[Y \mid A] \geq \mathbf{E}[X(\Delta)] \geq 2^8 \cdot \frac{\mathbf{B}(G)\Delta}{m} \cdot \frac{m}{\Delta} = 256 \cdot \mathbf{B}(G).$$

Then, by law of total probability and Lemma 20c, we have

$$\begin{aligned}
\Pr[Y \leq 16 \cdot \mathbf{B}(G)] &= \Pr[Y \leq 16 \cdot \mathbf{B}(G) \mid \bar{A}] \cdot \Pr[\bar{A}] \\
&\quad + \Pr[Y \leq 16 \cdot \mathbf{B}(G) \mid A] \cdot \Pr[A] \\
&\leq \frac{1}{2} + \frac{1}{2} \Pr[X(\Delta) \leq 1/16 \cdot \mathbf{E}[X(\Delta)]] \\
&\leq \frac{1}{2} + \frac{1}{2} \cdot \left( \frac{1}{4} + 2^{1-h/16} \right) \\
&\leq \frac{5}{8} + 2^{-h/16} \leq \frac{3}{4},
\end{aligned}$$

as  $h \geq 48$  for large enough  $n$ . Thus,  $u$  is a leader by time  $t + X + 8 \cdot \mathbf{B}(G)$  with probability at most  $3/4 + 1/16 = 13/16$ . Since  $\Pr[X \geq 16 \cdot \mathbf{E}[X(d)]] \leq 1/16$ , the lemma follows by union bound.  $\square$

The second technical lemma leverages this to show a higher concentration result for eliminating all-but-one candidate from contention, assuming all nodes are in the elimination phase.

**Lemma 24** *Suppose all leader candidates have degree at least  $d$  and all nodes are in the elimination phase. Let*

$$t(d) = 16(\tau + 2) \cdot \log_{8/7} n \cdot (\mathbf{E}[X(d)] + \mathbf{B}(G)).$$

*If no node reaches level  $\alpha(\tau)L$  by time  $t + t(d)$ , then exactly one leader candidate remains at time step  $t + t(d)$  with probability at least  $1 - O(n^{-\tau})$ .*

**Proof** Let  $A(u, v)$  denote the event that at time  $t + t(d)$  both  $u$  and  $v$  are leader candidates. The claim follows by using Lemma 23 and the union bound over all  $u \neq v$ , as

$$\sum_{u \neq v} \Pr[A(u, v)] \leq \sum_{u \neq v} \left( \frac{7}{8} \right)^{(\tau+2) \cdot \log_{8/7} n} \leq n^2 / n^{\tau+2} \leq 1/n^\tau.$$

$\square$

Next, we provide an upper bound on the time when, with high probability, all nodes are in the elimination phase, and all nodes of small degree have become followers.

**Lemma 25** *There exist constants  $\lambda \geq 1$  and  $\gamma \geq 1$  such that at time  $\lambda L \cdot \mathbf{B}(G)$  the following holds with probability  $1 - O(n^{-\tau})$ :*

1. *all nodes are in the elimination phase, and*
2. *all nodes of degree at most  $\Delta/\gamma$  are followers.*

**Proof** Let  $\lambda_0 \geq 1$  be a constant such that  $c(\lambda_0) \geq \tau$ , where  $c(\lambda) = \lambda - 1 - \ln \lambda$ . Define  $S(d, \ell)$  as in Lemma 22 and define  $t$  as  $t = 8\lambda_0^2 \mathbf{E}[S(\Delta, L)]$ . Note that  $t \in O(L \cdot \mathbf{B}(G))$  by Lemma 22 and by choice of  $h$ . First we show that by time

step  $t$  at least one node has reached level  $L$  with probability  $1 - O(n^{-\tau})$ . Consider a node  $v$  with degree  $\Delta$ . If at step  $t$  it is not a leader, then there must exist another node at level at least  $L$  by Rule (2). Otherwise, by Lemma 22, after  $t$  steps this node completed  $L$  streaks with probability at least  $1 - O(n^{-\tau})$ . Then there is at least one node at level at least  $L$  by time  $t$  with probability  $1 - O(n^{-\tau})$ . Let  $u$  be such node.

Notice that once broadcast from  $u$  reaches all other nodes, all nodes will be at level at least  $L$ . The probability that the broadcast from  $u$  does not reach all other nodes after  $2\mathbf{B}(G)$  steps is at most  $1/2$  by Markov's inequality. Then the probability that broadcast from  $u$  reaches all other nodes by time  $\mathbf{B}(G) \cdot L$  is at least  $1 - (1/2)^{\tau \log n} \in 1 - O(n^{-\tau})$  since  $L \geq 2\tau \log n$ . Thus, all nodes will be at level at least  $L$  within  $t + \mathbf{B}(G) \cdot L$  steps with probability  $1 - O(n^{-\tau})$ . As  $t \in O(L \cdot \mathbf{B}(G))$ , for large enough  $n$ , we have that  $t + \mathbf{B}(G) \cdot L \leq \lambda L \cdot \mathbf{B}(G)$  for some constant  $\lambda \geq 1$ . This completes the proof of the claim (1).

For claim (2), choose a constant  $\lambda_1 \leq 1$  such that  $c(\lambda_1) \geq \tau + 1$  and let  $\gamma = 32\lambda_0^2/\lambda_1^2$ . Clearly,  $\gamma$  is a constant at least 1, since  $\lambda_0 \geq 1$  and  $\lambda_1 \leq 1$  are constants. Consider a node  $w$  with degree  $\deg(w) = d \leq \Delta/\gamma$ . Let  $S \sim S(d, L)$  be the number of steps until  $w$  completes  $L$  streaks. Then by Lemma 22 we have that

$$\begin{aligned}
\Pr[S \leq t] &= \Pr[S \leq 8\lambda_0^2 \mathbf{E}[S(d, L)]] \leq \Pr[S \leq 8\lambda_0^2/\gamma \mathbf{E}[S]] \\
&= \Pr[S \leq \lambda_1^2/4 \mathbf{E}[S]] \leq 2/n^{c(\lambda_1)} \leq 2/n^{\tau+1}.
\end{aligned}$$

Claim (2) now follows by the union bound over all nodes with degrees less than  $\Delta/\gamma$ .  $\square$

Finally, we put everything together to obtain a bound on the time by which there is a single leader candidate left that holds with high probability.

**Lemma 26** *There exist constants  $\alpha(\tau)$  and  $C = C(\tau)$  such that there is exactly one leader candidate at time step  $C \cdot \mathbf{B}(G) \cdot \log n$  with probability at least  $1 - O(n^{-\tau})$ .*

**Proof** Let  $\lambda$  and  $\gamma$  by constants given by Lemma 25 and  $d = \lceil \Delta/\gamma \rceil$ . Then with probability  $1 - O(n^{-\tau})$  before  $\lambda L \cdot \mathbf{B}(G)$  steps all nodes are at level at least  $L$  and all nodes with degrees less than  $d$  are followers by Lemma 25. For the remaining part of the proof, suppose this holds. We now lower bound the time it takes for some node to reach the maximum level  $\alpha(\tau)L$ . Note that by Lemma 20 we get that

$$\begin{aligned}
\mathbf{E}[X(d)] &= (2^{h+1} - 2) \cdot \frac{m}{d} \\
&\leq 2^{h+1} \cdot \frac{m}{d} \\
&\leq 512 \cdot \mathbf{B}(G) \cdot \frac{\Delta}{m} \cdot \frac{m}{d} \leq 512\gamma \cdot \mathbf{B}(G).
\end{aligned}$$

Let  $\tau' \leq 1$  be a constant such that  $c(\tau') \geq \tau + 1$ . Set  $t$  as

$$\begin{aligned} t &= 16(\tau + 2)(512\gamma + 1) \cdot \log_{8/7} n \cdot \mathbf{B}(G) \\ &\geq 16(\tau + 2) \cdot \log_{8/7} n \cdot (\mathbf{E}[X(d)] + \mathbf{B}(G)) \end{aligned}$$

and set  $t'$  as

$$t' = 4096(\tau')^2 \cdot \mathbf{B}(G) \cdot (\alpha(\tau) - 1)L.$$

We can pick  $\alpha(\tau)$  to be a large enough constant such that  $t' > t$  holds for all large enough  $n$ , since both  $t$  and  $t'$  are of order  $\Theta(\mathbf{B}(G) \cdot \log n)$ .

Suppose there is a node  $v$  that reached level  $\alpha(\tau)L$ . We aim to show this happened after time  $\lambda L \cdot \mathbf{B}(G) + t'$  with probability  $1 - O(n^{-\tau})$ . As we mentioned above, we assume all nodes reached level  $L$  before time  $\lambda L \cdot \mathbf{B}(G)$ . After that, to reach level  $\alpha(\tau)L$  node  $v$  needs to complete a streak at least  $(\alpha(\tau) - 1)L$  times. Let  $S$  be the time for that to happen. Then  $S(\Delta, (\alpha(\tau) - 1)L) \leq S$ . Then by Lemma 22

$$\begin{aligned} \Pr[S \leq t'] &\geq \Pr[S(\Delta, (\alpha(\tau) - 1)L) \leq t'] \\ &\geq \Pr[S(\Delta, (\alpha(\tau) - 1)L) \leq (\tau')^2 \\ &\quad \cdot 8 \mathbf{E}[S(\Delta, (\alpha(\tau) - 1)L)]] \\ &\geq 1 - n^{-c(\tau')} \in 1 - O(n^{-\tau-1}). \end{aligned}$$

Then, by union bound, in  $t'$  steps none of the nodes have reached the maximal level  $\alpha(\tau)L$  with probability  $1 - O(n^{-\tau})$ . Since  $t' > t$ , Lemma 24 implies that a unique leader will be elected before time  $t$  with probability  $1 - O(n^{-\tau})$ . It remains to note that there exists a constant  $C(\tau)$  such that  $t < C(\tau)\mathbf{B}(G) \log n$  since  $L \in \Theta(\log n)$ .  $\square$

Finally, to guarantee finite expected stabilization time, the protocol includes a backup phase following the same approach as in [27]. The first node to reach level  $\alpha(\tau)L$  must be a leader candidate. When a node  $v$  reaches level  $\alpha(\tau)L$ , it switches to executing the constant-state token-based leader election protocol. When this happens, node initializes the constant-state protocol with the input  $\text{status}(v) \in \{\text{leader}, \text{follower}\}$  and starts running the protocol while simultaneously continues broadcasting its  $\text{level}(v)$  value using Rule (3). Within  $\mathbf{B}(G)$  expected steps, all nodes are running the constant-state protocol. This protocol guarantees that eventually only one leader remains after polynomially many expected steps.

**Theorem 5** *For any connected graph  $G$  with maximum degree  $\Delta$ , there is a leader election protocol that uses  $O(\log n \cdot h(G))$  states and stabilizes in  $O(\mathbf{B}(G) \cdot \log n)$  steps in expectation and with high probability, where  $h(G) \in O(\log(\Delta/\beta \cdot \log n))$  and  $\beta$  is the edge expansion of  $G$ .*

**Proof** The protocol uses  $O(hL)$  states. By Theorem 1,  $\mathbf{B}(G) \leq Cm/\beta \log n$  for some constant  $C$ . Thus,

$$\begin{aligned} h &= 8 + \left\lceil \log \left( \frac{\mathbf{B}(G) \cdot \Delta}{m} \right) \right\rceil \leq 8 \\ &\quad + \left\lceil \log \left( \frac{Cm \log n \cdot \Delta}{m\beta} \right) \right\rceil \in O \left( \log \left( \frac{\log n \cdot \Delta}{\beta} \right) \right). \end{aligned}$$

Clearly,  $L \in \Theta(\log n)$ , so the claim on the state complexity follows. By Lemma 26, the fast protocol stabilizes in  $O(\mathbf{B}(G) \cdot \log n)$  steps with probability at least  $1 - O(n^{-\tau})$ . By Theorem 3, the constant-state backup protocol stabilizes in  $O(n^4 \log n)$  expected steps, as the worst-case hitting time of a classic random walk is  $O(n^3)$  steps [32]. With probability at most  $O(n^{-\tau})$ , at least two leader candidates enter the backup phase. Choose  $\tau \geq 4$  and let  $T$  be stabilization time of the protocol. Then

$$T \in O \left( \mathbf{B}(G) \cdot \log n + n^{-\tau} \cdot n^4 \log n \right),$$

which is  $O(\mathbf{B}(G) \cdot \log n)$ .  $\square$

## 6 Time lower bounds for general protocols

In this section, we establish time lower bounds for stable leader election for general protocols with unbounded state space. First, we give a fairly general technique for constructing graphs, where leader election has a given time complexity of any order between  $\Omega(n \log n)$  and  $O(n^3)$ . This technique can be also applied to specific graph families to characterize the complexity of leader election in these families. Finally, we also give a result that shows that in any sufficiently dense graph leader election requires  $\Omega(n \log n)$  expected steps.

### 6.1 The lower bound construction for renitent graphs

We first introduce the notion of *isolating covers*. The idea is that we can cover the nodes of the graph with at most  $K$  subsets of the same size, each of which has isomorphic neighbourhood up to some distance  $\ell \geq 0$ , and that there are at least two such sets that are sufficiently far apart.

Let  $G = (V, E)$  be a graph and  $\mathcal{C} = \{V_0, \dots, V_{K-1}\} \subseteq 2^V$  be a collection of subsets of  $V$ . We say that  $\mathcal{C}$  is a  $(K, \ell)$ -cover of the graph  $G$  if

1. for each  $0 \leq i < j < K$  there exists an isomorphism  $\phi$  between  $G[B_\ell(V_i)]$  and  $G[B_\ell(V_j)]$  such that  $\phi(V_i) = V_j$ ,
2. there exists some  $V_i$  and  $V_j$  such that  $B_\ell(V_i) \cap B_\ell(V_j) = \emptyset$ , and
3.  $V_0 \cup \dots \cup V_{K-1} = V(G)$ .

That is, (1) the local neighbourhoods are isomorphic up to distance  $\ell$  and this isomorphism maps vertices of  $V_i$  to  $V_j$ , (2) there are two sets whose vertices are all far apart, and (3) the union of the sets covers the entire graph.

We define

$$Y(\mathcal{C}) = \min\{t : I_t(V_i) \setminus B_\ell(V_i) \neq \emptyset \text{ for some } V_i\}$$

to be the *isolation time* of the cover  $\mathcal{C}$ . This is the minimum time until some node in  $V_i$  is influenced by some node at distance greater than  $\ell$  from all nodes of  $V_i$ . We say that  $\mathcal{C}$  is  $t$ -*isolating* if  $\Pr[Y(\mathcal{C}) \geq t] \geq 1/2$ . This property states that it is unlikely that during the first  $t$  steps, nodes in the set  $V_i$  can be influenced by nodes that are far away from nodes in  $V_i$ .

Note that if the distance- $\ell$  propagation time on  $G$  satisfies  $\Pr[T_\ell(G) < t] \leq 1/2$ , then any  $(K, \ell)$ -cover of  $G$  is  $t$ -isolating. Thus, we may bound the minimum propagation times to show that a cover is isolating.

Let  $\mathcal{G}$  be an infinite family of graphs and  $f: \mathbb{N} \rightarrow \mathbb{N}$  be an increasing function. We say that graphs in  $\mathcal{G}$  are *f-rentent* if there exists a constant  $K \geq 2$  and function  $\ell: \mathbb{N} \rightarrow \mathbb{N}$  and such that every  $n$ -node graph  $G \in \mathcal{G}$  has an  $f(n)$ -isolating  $(K, \ell(n))$ -cover. In this section, we prove the following result.

**Theorem 6** *If the graph  $G$  is  $f$ -rentent, then any leader election protocol takes  $\Omega(f)$  expected steps to stabilize on  $G$ .*

Our approach is inspired by the lower bound construction for randomized leader election in synchronous message-passing models by Kutten et al. [29, Theorem 3.13]. However, unlike in synchronous message-passing models, in the population model communication is both stochastic and asynchronous with sequential interactions, so we need to further refine the approach to make it work in our setting.

We prove our result in a stronger variant of the population model: we do not restrict the number of states used by the nodes and give each node access to its own (independent and infinite) sequence of random bits. Formally, we assume that each node  $v \in V$  is given as input a random value  $y(v)$  sampled independently and uniformly at random from the unit interval  $[0, 1]$ . Since we do not restrict the state space of the nodes, the nodes can locally store this value to access an infinite sequence of i.i.d. random bits. The random bits assign nodes unique identifiers with probability 1. Any protocol that does not use these random bits can ignore them.

**Proof of Theorem 6** Suppose  $G$  is  $f$ -rentent and  $\mathcal{A}$  is leader election protocol on  $G$  that stabilizes in  $T$  steps. Without loss of generality, assume  $f(n) \geq 6$ , as otherwise the claim of Theorem 6 is trivially true.

Fix any  $f(n)$ -isolating  $(K, \ell)$ -cover  $\mathcal{C} = \{V_0, \dots, V_{K-1}\}$  of the graph  $G$ , where  $K$  is a constant independent of  $n$ , and

let  $Y = Y(\mathcal{C})$  be the isolation time of the cover. Let  $X \sim \text{Poisson}(\lambda)$  be a Poisson random variable with mean  $\lambda = f(n)/2$  and  $\mathcal{E}$  be the event that  $X < Y$ . Here,  $X$  represents a random time step (independent of  $Y$ ) at which we investigate the state of the system. To this end, we define  $L_i$  to be the event that some node  $v \in V_i$  outputs that it is a leader at step  $X$ .

**Lemma 27** *The following hold:*

- (a)  $\Pr[L_0 \mid \mathcal{E}] = \Pr[L_i \mid \mathcal{E}] \text{ for each } 0 \leq i < K, \text{ and}$
- (b)  $\Pr[L_i \cap L_j \mid \mathcal{E}] = \Pr[L_i \mid \mathcal{E}] \cdot \Pr[L_j \mid \mathcal{E}] \text{ for some } 0 \leq i < j < K.$

**Proof** Let  $P(G) = \{(v, u), (u, v) : \{v, u\} \in E(G)\}$  be the set of ordered pairs of nodes that can interact and let  $(e_t)_{t \geq 1}$  be a stochastic schedule on  $G$ . Note that each  $e_t$  is sampled from  $P$  uniformly at random independent of other interaction pairs. Let  $\sigma = (e_1, \dots, e_X)$  be the sequence of first  $X$  interactions, where  $X \sim \text{Poisson}(\lambda)$  is a Poisson random variable with mean  $\lambda = f(n)/2$ .

First we note that the distribution of  $\sigma$  can be equivalently expressed using the following continuous-time process. Suppose each pair  $(v, u) \in P(G)$  is activated at unit rate *independently* of all other elements of  $P(G)$  and previous activations of  $(v, u)$ , that is, the number of activations of  $(v, u)$  on time interval  $[0, t)$  is a Poisson random variable with mean  $t$ . The sum  $Z = Z_1 + Z_2$  of two independent Poisson random variables with mean  $\lambda_1$  and  $\lambda_2$  is a Poisson random variable with mean  $\lambda_1 + \lambda_2$ . Therefore, since  $|P(G)| = 2m$ , the total number  $Z$  of activated pairs during the interval  $[0, \lambda/(2m))$  is Poisson random variable with mean  $\lambda$ . That is,  $Z \sim X$  and the sequence  $\sigma' = (e'_1, \dots, e'_X)$ , where  $e'_i$  is the  $i$ th pair activated in the continuous-time process, has the same distribution as  $\sigma$ .

For each  $0 \leq i < K$ , let  $U_i = B_\ell(V_i)$  and  $\sigma_i$  be the sequence of pairs of nodes from  $G[U_i]$  that are activated during the time interval  $[0, \lambda/(2m))$  in the continuous time process. If the edge sets of  $G[U_i]$  and  $G[U_j]$  are disjoint, then  $\sigma_i$  and  $\sigma_j$  are independent. Moreover, since  $G[U_i] \simeq G[U_j]$  for all  $0 \leq i \leq j < K$ , the sequences  $\sigma_0, \dots, \sigma_{K-1}$  are identically distributed up to isomorphism, i.e.,  $\Pr[\sigma_i = a] = \Pr[\sigma_j = \phi(a)]$ , where  $\phi$  is the isomorphism between  $G[U_i]$  and  $G[U_j]$ . Note that  $\sigma_i$  corresponds to the longest subsequence of  $(e_1, \dots, e_X)$  of pairs of nodes which are both in  $U_i$ .

Now suppose the event  $\mathcal{E}$  occurs. Then the set of influencers satisfies  $I_X(V_i) \subseteq U_i$  for each  $0 \leq i < K$ . Property (1) of  $(K, \ell)$ -covers implies that  $|V_i| = |V_j|$  for each  $i \neq j$ . Let  $N = |V_0|$  and  $x_{i,t} : V_i \rightarrow \Lambda$  be the configuration of nodes in the set  $V_i$  at after  $t$  steps. If  $\mathcal{E}$  occurs, then  $x_{i,t}$  can only depend on the sequence  $\sigma_i$  of interactions between nodes in  $U_i$  and the initial random values  $y_i \in [0, 1]^{U_i}$  assigned to

nodes in  $U_i$ . In particular, we get that  $x_{i,X}$  is a function of the pair  $(y_i, \sigma_i)$ .

Since the isomorphism  $\phi$  takes  $V_i$  to  $V_j$ ,  $(y_i, \sigma_i)$  and  $(y_j, \sigma_j)$  are identically distributed (but not necessarily independent). Hence, the configurations  $x_{i,X}$  and  $x_{j,X}$  are identically distributed given that  $\mathcal{E}$  occurs. This implies claim (a) of the lemma. By the second property of  $(K, \ell)$ -covers, there exist some  $i < j$  such that  $B_\ell(V_i) \cap B_\ell(V_j) = \emptyset$  and so  $U_i \cap U_j = \emptyset$ . If the event  $\mathcal{E}$  happens, then edges of  $G[U_i]$  and  $G[U_j]$  are disjoint, and so the random sequences  $\sigma_i$  and  $\sigma_j$  are conditionally independent given  $\mathcal{E}$ . Thus,  $(y_i, \sigma_i)$  and  $(y_j, \sigma_j)$  are also conditionally independent given  $\mathcal{E}$ , which implies that  $x_{i,X}$  and  $x_{j,X}$  are conditionally independent given  $\mathcal{E}$ . This implies claim (b).  $\square$

**Lemma 28** *There exists a constant  $C(K) > 0$  such that the stabilization time  $T$  of protocol  $\mathcal{A}$  satisfies  $\Pr[T > X] \geq C(K)$ .*

**Proof** Let  $x$  be the configuration at the random time step  $X$ . Note that  $T \leq X$  if and only if  $x$  is a stable configuration with a unique leader. Let  $A$  denote the event  $T \leq X$  and define  $\Pr[\bar{A}] = \alpha$ ,  $\Pr[\mathcal{E}] = \epsilon$  and  $\Pr[L_0 \mid \mathcal{E}] = \gamma$ . Observe that

$$\begin{aligned} \Pr[A \mid \mathcal{E}] &= \frac{\Pr[A \cap \mathcal{E}]}{\Pr[\mathcal{E}]} = \frac{\Pr[A] - \Pr[A \mid \bar{\mathcal{E}}] \cdot \Pr[\bar{\mathcal{E}}]}{\Pr[\mathcal{E}]} \\ &\geq \frac{1 - \alpha - (1 - \epsilon)}{\epsilon} = 1 - \frac{\alpha}{\epsilon}. \end{aligned}$$

Since the configuration  $x$  contains a node in a leader state, union bound yields

$$\Pr[A \mid \mathcal{E}] \leq \sum_{i=0}^{K-1} \Pr[L_i \mid \mathcal{E}] = K\gamma,$$

where the equality of probabilities follows from Lemma 27a. Let  $i < j$  be as given by Lemma 27b. Since the configuration  $x$  cannot be stable if at least two nodes are in a leader state, we have that

$$\Pr[\bar{A} \mid \mathcal{E}] \geq \Pr[L_i \cap L_j \mid \mathcal{E}] = \Pr[L_i \mid \mathcal{E}] \cdot \Pr[L_j \mid \mathcal{E}] = \gamma^2,$$

because by Lemma 27b the events  $L_i$  and  $L_j$  are conditionally independent given  $\mathcal{E}$ . Combining all of the above yields

$$1 - \alpha/\epsilon \leq \Pr[A \mid \mathcal{E}] \leq K\gamma \leq K\sqrt{\alpha/\epsilon}.$$

Thus,  $\alpha$  needs to satisfy the quadratic inequality  $1 - \alpha/\epsilon \leq K\sqrt{\alpha/\epsilon}$ . The solution of this inequality for  $\epsilon > 0$  is given by

$$\sqrt{\alpha} \geq \frac{\sqrt{\epsilon}}{2} \left( \sqrt{K^2 + 4} - K \right) \geq 0.$$

Finally, we show that  $\epsilon = \Pr[\mathcal{E}] > 1/4$ . Since the cover  $V_0, \dots, V_{K-1}$  is  $f(n)$ -isolating, we have  $\Pr[Y \geq f(n)] \geq 1/2$ . By the law of total probability,

$$\begin{aligned} \Pr[Y < X] &\leq \Pr[X > 2\lambda] + \Pr[Y \leq 2\lambda] \\ &\leq \exp(-f(n)/4) + \Pr[Y \leq f(n)] \\ &\leq \exp(-3/2) + 1/2 \leq 3/4, \end{aligned}$$

where we applied the definition of  $\lambda = f(n)/2$ , Lemma 1a with  $c = 2$ , and the assumption  $f(n) \geq 6$ . Thus,  $\alpha > 0$  is a positive constant denoting the probability of the event  $T > X$ .  $\square$

**Theorem 6** *If the graph  $G$  is  $f$ -renitent, then any leader election protocol takes  $\Omega(f)$  expected steps to stabilize on  $G$ .*

**Proof** We first show that there exist constants  $n_0$  and  $C < 1$  such that for any  $n \geq n_0$ , we have  $\Pr[T \leq \lambda/2] \leq C$ . By law of total probability and the definition of conditional probability, we get that

$$\begin{aligned} \Pr[T \leq \lambda/2] &\leq \Pr[T \leq X] + \Pr[T \leq \lambda/2 \wedge T > X] \\ &\leq \Pr[T \leq X] + \Pr[X \leq \lambda/2] \\ &\leq 1 - C(K) + \exp(-f(n)/12) \leq C. \end{aligned}$$

We can pick constant  $C < 1$  satisfying the above since  $C(K) > 0$  is a constant by Lemma 28 and  $\exp(-f(n)/12)$  is a term that tends to 0 as  $n$  increases, as  $f$  is an increasing function. Finally, since  $1 - C > 0$  is a positive constant and  $T$  is a non-negative random variable, we have

$$\begin{aligned} \mathbf{E}[T] &\geq \mathbf{E}[T \mid T \geq \lambda/2] \cdot \Pr[T \geq \lambda/2] \\ &\geq \frac{(1 - C)\lambda}{2} = \frac{1 - C}{4}f(n). \end{aligned}$$

$\square$

## 6.2 Constructing renitent graphs

We now give examples of  $f$ -renitent graphs; by Theorem 6 the expected stabilization time on these graphs will be  $\Omega(f)$ . For example, as a warmup, it is not hard to see that cycles are  $\Omega(n^2)$ -renitent: We can split the cycle into six paths  $V_0, \dots, V_5$  of length roughly  $n/6$  and information propagation from set  $V0$  to  $V3$  requires  $\Omega(n^2)$  steps with constant probability.

**Lemma 29** *Cycle graphs are  $\Omega(n^2)$ -renitent.*

**Proof** Let  $G$  be an  $n$ -cycle with nodes  $V = \{v_0, \dots, v_{n-1}\}$  and  $n = 6\ell$ . Define

$$V_i = \{v_{i\ell}, \dots, v_{(i+1)\ell-1}\}$$

for  $i \in \{0, 1, \dots, 5\}$ . Observe that each  $B_\ell(V_i)$  is isomorphic to a path of the same length,  $B_{\ell-1}(V_0) \cap B_{\ell-1}(V_3) = \emptyset$  and  $V_0 \cup \dots \cup V_5 = V$ . Lemma 13 implies that

$$\Pr[T_\ell(G) \leq c\ell n] \leq 1/n \leq 1/2$$

for some constant  $c > 0$ . Thus,  $\{V_0, \dots, V_5\}$  is  $\Omega(n^2)$ -isolating  $(6, \ell)$ -cover.  $\square$

In fact, for any constant  $k > 0$ , the above idea generalizes to higher dimensions:  $k$ -dimensional toroidal grids are  $\Omega(n^{1+1/k})$ -renitent; one can partition such grids into constantly many subcubes of diameter  $\Theta(n^{1/k})$  and observe that propagating information to distance  $\Theta(n^{1/k})$  in regular graphs requires  $\Omega(n^{1+1/k})$  steps with constant probability.

The next lemma allows us to obtain  $\Omega(Dm)$ -renitent graphs for essentially any diameter  $D$  and number  $m$  of edges.

**Lemma 30** *Let  $G$  be a connected graph with  $n$  nodes,  $m$  edges and diameter  $D$ . For any integer  $\ell$  such that  $D \leq \ell \leq n$ , there exists an  $\Omega(\ell m)$ -renitent graph  $G'$  with  $\Theta(n)$  nodes,  $\Theta(m)$  edges and diameter  $\Theta(\ell)$ . In addition,  $\mathbf{B}(G') \in \Omega(\ell m)$ .*

**Proof** Fix a node  $v^*$  of  $G$ . We construct the graph  $G'$  as follows: take four copies  $G_0, \dots, G_3$  of the graph  $G$  and connect the  $i$ th copy  $v_i^*$  of  $v^*$  to  $v_{(i+1) \bmod 4}^*$  by using a path  $P_i$  of length  $2\ell$ . Define  $V_i = V(G_i) \cup V(P_i)$  for  $i \in \{0, 1, 2, 3\}$ . By construction  $\mathcal{C} = \{V_0, \dots, V_3\}$  gives a  $(4, \ell)$ -cover of  $G'$ , as  $G'[B_k(V_i)] \simeq G'[B_k(V_j)]$  for all  $i, j \in \{0, 1, 2, 3\}$  and  $k \geq 0$ ,  $B_\ell(V_0) \cap B_\ell(V_2) = \emptyset$  and  $V_0 \cup \dots \cup V_3 = V(G')$ .

We now show that the cover is  $t$ -isolating for  $t = \lambda\ell m$  for some constant  $\lambda > 0$ . For each  $0 \leq i \leq 3$ , let  $\rho_i$  and  $\rho'_i$  be the sequence of edges in  $P_{(i+1) \bmod 4}$  and  $P_{(i-1) \bmod 4}$  ordered towards  $v_i^*$ . Note that for  $I_t(V_i) \setminus B_\ell(V_i) \neq \emptyset$  to hold, the scheduler must have sampled either at least half of sequence  $\rho_i$  or at least half of  $\rho'_i$  (both sequences have length  $2\ell$ ), since  $D \leq \ell$ . Denote by  $\hat{\rho}_i$  and  $\hat{\rho}'_i$

the halves of  $\rho_i$  and  $\rho'_i$  respectively that are closest to  $V_i$ . Now for any  $0 \leq i \leq 3$ , we have that

$$\begin{aligned} \Pr[I_t(V_i) \setminus B_\ell(V_i) \neq \emptyset] &\leq \Pr[X(\hat{\rho}_i) > t] + \Pr[X(\hat{\rho}'_i) > t] \\ &\leq 2 \cdot \exp(-\ell c(\lambda)) \leq 1/8 \end{aligned}$$

by using Lemma 5 and choosing  $\lambda > 0$  to be a sufficiently small constant. By union bound,  $\Pr[Y(\mathcal{C}) \geq \lambda\ell m] \geq 1/2$ . The above also implies that the distance  $\ell$ -propagation time of  $v_i^*$  satisfies  $\Pr[T_\ell(v_i^*) > t] \geq 7/8$ . It now follows that  $\mathbf{B}(G) = \max\{\mathbf{E}[T(u)] : u \in V\} \in \Omega(\ell m)$ , as

$$\mathbf{E}[T(v_i^*)] \geq \mathbf{E}[T_\ell(v_i^*) \mid T_\ell(v_i^*) > t] \cdot \Pr[T_\ell(v_i^*) > t] \geq \frac{7t}{8}.$$

**Theorem 7** *For any increasing function  $T : \mathbb{N} \rightarrow \mathbb{N}$  such that  $n \log n \leq T(n) \leq n^3$ , there is an infinite family of graphs in which stable leader election takes  $\Theta(T(n))$  expected steps and the broadcast time satisfies  $\mathbf{B}(G) \in \Theta(T)$ .*

**Proof** For any  $N \geq 1$ , we construct a graph  $G$  with  $n \geq N$  nodes as follows. We distinguish two cases:

- First, if  $T \in \omega(n^2 \log n)$ , then take a clique  $H$  of size  $N$  and set  $\ell = \lceil T(N)/N^2 \rceil$ .
- Otherwise, if  $T \in O(n^2 \log n)$ , then set  $\ell = \lceil \log N + T(N)/(N \log N) \rceil$ , take a star graph and add  $\Theta(T(N)/\ell)$  edges in an arbitrary fashion to obtain the graph  $H$ . Adding this many edges is always possible since  $T(N)/\ell \in O(N^2)$ .

In both cases, we apply Lemma 30 with  $H$  and  $\ell$  to obtain a graph  $G$ . Note that Lemma 30 implies that the graph  $G$  will be  $\Omega(T)$ -renitent and satisfy  $\mathbf{B}(G) \in \Omega(T)$ . By Theorem 6 stable leader election will take  $\Omega(T)$  expected steps on this graph. The graph  $H$  has constant diameter, and since  $\ell \in \Omega(\log n)$ , the graph  $G$  has diameter  $\Omega(\log n)$ . By construction,  $Dm \in \Theta(T)$ , so Theorem 1 implies that  $\mathbf{B}(G) \in O(T)$ . Now Theorem 4 implies the upper bound for leader election time.  $\square$

### 6.3 A lower bound for dense graphs

The above construction gives graph families in which expected leader election and broadcast time are of the same order. However, this is not generally true. Leader election time can be much lower than broadcast time in graphs, where the local structure helps break symmetry fast. The star graph (i.e., a tree of depth one) is the simplest example: there is a trivial constant-state protocol that elects a leader in one interaction, but broadcast time in a star is  $\Theta(n \log n)$  by a simple coupon collector argument.

The above example rules out, for example, the existence of a general  $\Omega(n \log n)$  lower bound for leader election in sparse graphs. In this section, we show that in dense graphs with sufficiently high minimum degrees, we cannot easily exploit local graph structure to break symmetry fast, even if we use any number of states per node.

**Theorem 8** *Let  $0 < \lambda < 1$  and  $0 < \phi < 1$  be constants. If  $G$  has minimum degree  $\delta \geq \lambda n^\phi$  and  $m \geq \lambda n^2$  edges, then any stable leader election protocol requires  $\Omega(n \log n)$  expected steps to stabilize on  $G$ .*

At its core, the argument is an extension of lower bound result of Sudo and Masuzawa [17] from cliques to general high-degree graphs. However, to deal with the general structure of the interaction graph, we introduce the two new

concepts: multigraphs of influencers and leader generating interaction patterns.

Our proof strategy is roughly as follows. We assume that there is a fast protocol that stabilizes in  $o(n \log n)$  steps in a graph with the properties as in the above theorem. First, we capture the spatial structure of the part of the graph that influences a node to be elected as a leader; we call such structures ‘‘leader generating interaction patterns’’. We show that there must be such patterns that are fairly small and almost tree-like. This means they can be unfolded into trees, without growing their size too much. Then we argue that, because the graph has high degrees, such a tree is likely to be found in the set of nodes that have not interacted by time  $o(n \log n)$ . Since a new leader can be generated in this part of the graph, this implies that any configuration reached in  $o(n \log n)$  steps is unlikely to be stable.

Recall that  $I_t(v) \subseteq V$  denotes the set of influencers of node  $v$  at time  $t$ . We start with the following lemma showing that the sets of influencers grow slowly on dense graphs.

**Lemma 31** *Let  $0 < \varepsilon < 1$  be a constant. There exists a constant  $0 < c < c(\varepsilon, \lambda)$  such that for any node  $v \in V$  and any  $0 \leq t \leq cn \log n$ , we have*

$$\Pr[|I_t(v)| > n^\varepsilon] \in \exp\left(-\Omega\left(\sqrt{n^\varepsilon}\right)\right).$$

**Proof** Define

$$\begin{aligned} K &= \left\lfloor \lambda/2 \cdot \left(n^{\varepsilon/2} - 1\right) \right\rfloor & \ell &= \left\lfloor 2/\lambda \cdot \left(n^{\varepsilon/2} - 1\right) \right\rfloor \\ r_k &= \left\lfloor \frac{\lambda n}{2k} \right\rfloor & t_0 &= \sum_{k=1}^K r_k. \end{aligned}$$

Note that  $K\ell \leq n^\varepsilon - 1$ . Assume that  $n$  is large enough so that  $t_0 > 0$ . Let  $J_0(v) = \{v\}$ , and for  $0 \leq t < t_0$  define

$$J_{t+1}(v) = \begin{cases} J_t(v) \cup \{w\} & \text{if } e_{t_0-t} = (u, w) \text{ or } e_{t_0-t} = (w, u) \text{ for some } w \in J_v(t) \\ J_t(v) & \text{otherwise,} \end{cases}$$

where we take  $e_0 = \emptyset$ . That is,  $J_{t+1}(v) = J_t(v) \cup \{w\}$  if some node in  $u \in J_v(t)$  interacted with node  $w$  at step  $t_0 - t$ . We can think of  $J_t(v)$  as an evolution of  $I_t(v)$  played in reverse. Note that  $J_{t_0}(v) = I_{t_0}(v)$ . Hence, it is enough to show that the event  $|J_{t_0}(v)| < n^\varepsilon$  is likely to happen.

We now lower bound the number of steps it takes for  $J_t(v)$  to grow from 1 to  $K\ell + 1 \leq n^\varepsilon$ . Note that  $J_t(v)$  changes by at most one every step. Moreover,

$$\Pr[|J_{t+1}(v)| = k + 1 \mid |J_t(v)| = k] \leq \frac{nk}{m} \leq \frac{k}{\lambda n} = p(k),$$

as  $m \geq \lambda n^2$  and there are most  $n|J_t(v)|$  edges incident to nodes in  $J_t(v)$ . Let  $t(a, b)$  be the time it takes for  $|J_t(v)|$  to

grow from size  $a$  to size  $b$ . Note that when  $J_t(v)$  has size at most  $b$ , then the probability of  $J_t(v)$  growing in a single step is upper bounded by  $p(b) = b/(\lambda n)$ .

We divide the interval  $[1, K\ell + 1]$  into  $K$  disjoint intervals of length  $\ell$ . Let  $[a_k, b_k]$  to be  $k$ th such interval. For the count to grow from  $a_k$  to  $b_k$  we need  $b_k - a_k = \ell$  steps in which the size of  $J_t(v)$  increases. The probability that it takes no more than  $r_k$  steps for the count to grow from  $a_k$  to  $b_k$  can be upper bounded by the probability of the event  $Y_k \geq \ell$ , where  $Y_k \sim \text{Bin}(r_k, p(b_k))$  is a binomial random variable. Since

$$\mathbf{E}[Y_k] = r_k \cdot p(b_k) \leq \frac{\lambda n}{2k} \cdot \frac{b_k}{\lambda n} \leq \frac{\ell k}{2k} = \ell/2,$$

the Chernoff bound from Lemma 2 implies that

$$\begin{aligned} \Pr[t(a_k, b_k) \leq r_k] &\leq \Pr[Y_k \geq \ell] \\ &\leq \Pr[Y_k \geq 2 \cdot \mathbf{E}[Y_k]] \in \exp\left(-\Omega\left(\sqrt{n^\varepsilon}\right)\right), \end{aligned}$$

as  $\mathbf{E}[Y_k] \in \Theta(\sqrt{n^\varepsilon})$ . Now by using the union bound over all  $K$  intervals, we have

$$\begin{aligned} \Pr[|J_{t_0}(v)| \leq n^\varepsilon] &\leq \Pr[t(1, n^\varepsilon) \leq t_0] \\ &\leq \Pr\left[t(1, K\ell + 1) \leq \sum_{k=1}^K r_k\right] \\ &\leq \sum_{k=1}^K \Pr[t(a_k, b_k) \leq r_k] \\ &\in \exp\left(-\Omega\left(\sqrt{n^\varepsilon}\right)\right), \end{aligned}$$

since  $\mathbf{E}[Y_k], K, \ell \in \Theta(\sqrt{n^\varepsilon})$ . Now we observe that for all sufficiently large  $n$ , we have

$$\begin{aligned} t_0 &= \sum_{k=1}^K r_k \geq \sum_{k=1}^K \left(\frac{\lambda n}{2k} - 1\right) \\ &= \frac{\lambda n}{2} \left(\sum_{k=1}^K \frac{1}{k}\right) - K = \frac{\lambda n H_K}{2} - K \geq cn \log n, \end{aligned}$$

where  $H_K$  is the  $K$ th harmonic number and  $c > 0$  is some constant such that

$$c \log n \leq \frac{\lambda H_K}{2} - \frac{K}{n},$$

for all sufficiently large  $n$ . Such a constant exists, since  $K \in \Theta(\sqrt{n^\varepsilon})$  and  $0 < \varepsilon < 1$  is a constant. Thus, the claim of the lemma follows for this  $c$ .  $\square$

Next we show that for any set  $U$  of nodes of size  $N$  and any constant  $\varepsilon > 0$ , there likely is a subset of  $\Omega(N^{1-\varepsilon})$  nodes in  $U$  that have not interacted by time step  $o(n \log n)$ .

**Lemma 32** Let  $0 < \varepsilon \leq 1$  be a constant. Fix a set  $U \subseteq V$  of size  $N$ . Let  $X(t)$  be the number of nodes in  $U$  that have not interacted by step  $t \geq 0$ . Then for any small enough constant  $0 < c < c(\varepsilon, \lambda)$  and for all  $0 \leq t < cn \log N$ , we have

$$\Pr[X(t) \leq N^{1-\varepsilon}] \in \exp\left(-\Omega\left(N^{1-\varepsilon}\right)\right).$$

**Proof** Clearly  $X(0) = N$ . We estimate the time until  $X(t) < N^{1-\varepsilon}$  holds. Note that

$$\Pr[X(t+1) < X(t) \mid X(t) = k] \leq \frac{kn}{m} \leq \frac{k}{\lambda n} = p(k),$$

as  $m \geq \lambda n^2$ . Moreover,  $0 \leq X(t) - X(t+1) \leq 2$ , as in each iteration the number of nodes that have interacted can increase by at most two. Let  $t(a, b)$  denote the number of steps until  $X(t)$  decreases from  $a$  to  $b$ . Given that  $X(t) = a$ , the probability that the number of nodes that have not yet interacted decreases is at most  $p(a)$ . For their count to drop below  $b$ , we need at least  $(a-b)/2$  steps in which the count decreases.

Define

$$\ell = 2 \left\lceil N^{1-\varepsilon} \right\rceil \quad K = \left\lfloor \frac{N}{\ell} \right\rfloor - 1 \quad t_k = \left\lfloor \frac{\lambda n}{4(k+1)} \right\rfloor$$

$$t = \sum_{k=1}^K t_k$$

and partition the interval  $[(K+1)\ell, \ell]$  into  $K$  disjoint intervals of length  $\ell$ . Let  $[a_k, b_k]$  be the  $k$ th interval, where  $a_k = (k+1)\ell$  and  $b_k = k\ell$ .

The probability of the event  $t(a_k, b_k) \leq t_k$  is at most the probability of the event  $Y_k \leq (a_k - b_k)/2 = \ell/2$ , where  $Y_k \sim \text{Bin}(t_k, p(a_k))$  is a binomial random variable. From the Chernoff bound of Lemma 2 and the fact that  $\mathbf{E}[Y_k] \leq \ell/4$ , we get that

$$\Pr[t(a_k, b_k) \leq t_k] = \Pr\left[Y_k \geq \frac{\ell}{2}\right]$$

$$\leq \Pr[Y_k \geq 2 \cdot \mathbf{E}[Y_k]] \in \exp\left(-\Omega\left(N^{1-\varepsilon}\right)\right),$$

as  $\mathbf{E}[Y_k] \in \Theta(N^{1-\varepsilon})$ . Taking the union bound over all  $K \in \Theta(N^\varepsilon)$  intervals yields

$$\Pr[X(t) \leq N^{1-\varepsilon}] = \Pr\left[t(N, N^{1-\varepsilon}) \leq \sum_{k=1}^K t_k\right]$$

$$\leq \sum_{k=1}^K \Pr[t(a_k, b_k) \leq t_k]$$

$$\in \exp\left(-\Omega\left(N^{1-\varepsilon}\right)\right).$$

Now observe that for all sufficiently large  $N$ , we have

$$t = \sum_{k=1}^K t_k \geq \sum_{k=1}^K \left( \frac{\lambda n}{4(k+1)} - 1 \right) \geq \frac{\lambda n}{4} \left( \sum_{k=1}^K \frac{1}{k+1} \right) - K$$

$$\geq \frac{\lambda n (H_K - 1)}{4} - K \geq cn \log N,$$

where  $H_K$  is the  $K$ th harmonic number and  $c > 0$  is some constant. Such a constant exists as  $K \in \Theta(N^\varepsilon)$  and  $0 < \varepsilon < 1$  is a constant.  $\square$

Now we show that the set of nodes that remain in their initial state after  $o(n \log n)$  steps is likely to contain a large induced subgraph that contains a polynomially-sized tree. Let  $S(t)$  denote the set of nodes that have not interacted by step  $t$ .

**Lemma 33** Suppose  $G$  has  $m \geq \lambda n^2$  edges and minimum degree  $\delta \geq \lambda n^\phi$ . There exist constants  $\varepsilon > 0$  and  $c > 0$  such that for all  $t \leq cn \log n$ , the subgraph induced by nodes in  $S(t)$  contains any tree of size  $n^{\varepsilon+c}$  with high probability.

**Proof** Fix  $\varepsilon < 1/2$  to be a small enough constant such that  $\phi(1-\varepsilon) > 2\varepsilon$ . Note that for all sufficiently large  $n$ , we have the following properties:

(a) By applying Lemma 32 with  $U = V$ , we get that there exists a positive constant  $c_1 = c_1(\varepsilon, \lambda)$  such that the event

$$|S(t)| \geq n^{1-\varepsilon}$$

happens with high probability for any  $0 \leq t \leq c_1 n \log n$ .

(b) Note that the minimum degree of  $G$  is at least  $\delta \geq \lambda n^\phi$  in  $G$ . For any  $v \in V$ , by applying Lemma 32 with the set  $U = B(v)$ , we get that there exists a positive constant  $c_2 = c_2(\varepsilon, \lambda)$  such that the event

$$|B(v) \cap S(t)| \geq |B(v)|^{1-\varepsilon} \geq \delta^{1-\varepsilon} \geq n^{\phi(1-\varepsilon)}$$

happens with high probability for any  $0 \leq t \leq c_2 n \log n$ .

By union bound, the event happens for all  $v \in V$  with high probability.

Let  $c = \min\{c_1, c_2, \varepsilon\}$  and  $t \leq cn \log n$ . Note that  $c$  is a positive constant. We condition the rest of the proof that the events in (a) and (b) happen for this  $t$ .

Assume that  $n$  is large enough so that  $n^{\phi(1-\varepsilon)} > 2n^{2\varepsilon} > 2n^{\varepsilon+c} = 2k$  holds and suppose  $T$  is any tree of size  $k = n^{\varepsilon+c}$ . We show that there is an isomorphic tree in the subgraph of  $G$  induced by  $S(t)$ . Let  $U = \{u_1, \dots, u_k\}$  be the nodes of the tree  $T$  ordered by a breadth-first search and  $U_i = \{u_1, \dots, u_i\}$ . We define  $T_i$  to be the subgraph of  $T$  induced by the nodes  $U_i$ . We show by induction that we can map

each node  $u_i \in V(T)$  to a node  $v_i \in V(G)$  such that the set  $\{v_1, \dots, v_i\}$  induces a subgraph in  $T$  that contains the edges of  $T_i$ .

For the base case, we map  $u_1$  to an arbitrary node  $v_1 \in S(t)$ . For the inductive step, suppose that  $\{v_1, \dots, v_i\}$  induces a subgraph that contains the edges of  $T_i$ . Clearly  $u_{i+1}$  has at most  $k$  neighbours in  $T$ , as  $T$  has at most  $k$  nodes. Let  $u_j \in U_i$  be the parent of  $u_{i+1}$  in  $T$ . Note that  $j \leq i$ . From (b), we have that

$$|S(t) \cap B(v_j)| - |U_i| \geq n^{\phi(1-\varepsilon)} - k > k,$$

so we can find a node  $v_{i+1} \in S(t) \cap B(v_j)$  that is not in  $\{v_1, \dots, v_i\}$ . Map  $u_{i+1}$  to this node. Now the subgraph induced by  $\{v_1, \dots, v_{i+1}\}$  contains the edges of  $T_{i+1}$ .  $\square$

### Multigraphs of influencers

To finish the proof, we need more knowledge about how the sets of influencers look like. For each  $v \in V$  and  $t$ , we construct a directed *multigraph of influencers*  $\mathcal{I}_t(v)$  that keeps tracks of the interactions that have influenced  $v$  in a more fine-grained manner than the set  $I_t(v)$  of influencers alone would. The latter only keeps track of the nodes that can influence the state of  $v$  at step  $t$ , not the sequence of interactions that have influenced the node.

We define this multigraph as follows. Let  $(e_t)_{t \geq 1}$  be the stochastic schedule on the interaction graph  $G$ . The set of nodes of  $\mathcal{I}_t(v)$  will be the set of influencers  $I_t(v)$  and the *multiset* of (directed) edges is given by the following recurrence. Define  $\mathcal{I}_t(v) = \emptyset$  and for all  $t \geq 0$ , define

$$\mathcal{I}_{t+1}(v) = \begin{cases} \mathcal{I}_t(v) \cup \mathcal{I}_t(u) \cup \{(v, u)_{t+1}\} & \text{if } e_{t+1} = (v, u) \\ \mathcal{I}_t(v) & \text{otherwise.} \end{cases}$$

In the first line,  $(v, u)_{t+1}$  is understood to be a directed edge timestamped with the time step  $t+1$  of its activation. That is,  $\mathcal{I}_t(v)$  is a *directed multigraph* where edges of multiplicity  $> 1$  are allowed as long as they have different timestamps. However, note that there can be no self-loops.

The multigraph  $\mathcal{I}_t(v)$  *fully* captures interactions that happened before time step  $t$  and which can affect the state of  $v$  at step  $t$ . Hence, given  $\mathcal{I}_t(v)$ , we can determine the state of node  $v$  at time  $t$ . This gives us more information than sets of influencers alone.

It will be easier to analyze  $\mathcal{I}_t(v)$ , when its evolution is played in reverse. Fix  $t_0 \geq 0$  and define  $\mathcal{J}_t(v)$  to be the induced subgraph constructed on edges of  $\mathcal{I}_t(v)$  that were added after step  $t_0 - t$  for  $t \leq t_0$ . Observe that  $\mathcal{J}_{t_0}(v) = \mathcal{I}_{t_0}(v)$  and that  $\mathcal{J}_t(v)$  follows the recurrence given

by  $\mathcal{J}_0(v) = (\{v\}, \emptyset)$  and

$$\mathcal{J}_{t+1}(v) = \begin{cases} \mathcal{J}_t(v) \cup \{(u, w)_{t_0-t}\} & \text{if } e_{t_0-t} = (u, w) \text{ and} \\ & \{u, w\} \cap V(\mathcal{J}_t(v)) \neq \emptyset \\ \mathcal{J}_t(v) & \text{otherwise.} \end{cases}$$

In the first line,  $V(\mathcal{J}_t(v))$  denotes the nodes of  $\mathcal{J}_t(v)$ , and  $\mathcal{J}_t(v) \cup \{(u, w)_{t_0-t}\}$  is interpreted as adding a directed edge to the multiset as edges of  $\mathcal{J}_t(v)$  along with adding  $u$  and  $w$  to the set of nodes of  $\mathcal{J}_t(v)$ , if one of them does not belong to  $V(\mathcal{J}_t(v))$  yet. We also keep original timestamps on edges of the multigraph.

Call an interaction of two nodes  $u$  and  $w$  at time step  $t - t_0$  *internal* if both  $u$  and  $w$  are contained in  $\mathcal{J}_v(t)$ . Internal interactions create cycles in  $\mathcal{J}_v(t)$ . We show that internal interactions almost never happen before  $\Omega(n \log n)$  steps, which means that  $\mathcal{J}_v(t_0)$  is tree-like for small enough  $t_0$ .

**Lemma 34** *For any  $\varepsilon > 0$ , any small enough  $0 < c < c(\varepsilon, \lambda)$  and any  $t_0 \leq cn \log n$ , with high probability the following events happen for all  $v \in V$  simultaneously*

- *there are at most  $c \log n$  internal interactions in  $\mathcal{J}_{t_0}(v)$ , and*
- *$\mathcal{J}_{t_0}(v)$  has size at most  $n^\varepsilon$ .*

**Proof** Consider some  $v \in V$ . By Lemma 31, for any small enough  $c$ , we have that

$\mathcal{I}_{t_0}(v)$  contains at most  $n^\varepsilon$  nodes with high probability for any  $t_0 \leq cn \log n$ . Hence,  $\mathcal{J}_{t_0}(v)$  also contains at most  $n^\varepsilon$  nodes. Thus, for all  $0 \leq t \leq t_0$  we know that  $\mathcal{J}_t(v)$  contains at most  $n^\varepsilon$  nodes, since the size of  $\mathcal{J}_t(v)$  does not exceed the size of  $\mathcal{J}_{t_0}(v)$ . Therefore, the probability that any two nodes in  $\mathcal{J}_t(v)$  interact during any time step  $1 \leq t \leq t_0$  is at most

$$p = \frac{n^{2\varepsilon}}{2m} \leq \frac{1}{\lambda n^{2-2\varepsilon}}.$$

The number of internal interactions during the interval  $\{1, \dots, t_0\}$  is stochastically dominated by a binomial random variable  $X \sim \text{Bin}(t_0, p)$  with mean  $\mathbf{E}[X] = t_0 p$ . Using the version of Chernoff's bound as in Theorem 2.3.1 of [48], we have

$$\begin{aligned} \Pr[X \geq c \log n] &\leq \exp(-\mathbf{E}[X]) \left( \frac{e \mathbf{E}[X]}{c \log n} \right)^{c \log n} \\ &\leq \left( \frac{e \mathbf{E}[X]}{c \log n} \right)^{c \log n} \\ &\leq \left( \frac{et_0 p}{c \log n} \right)^{c \log n} \leq \left( \frac{e}{\lambda n^{1-2\varepsilon}} \right)^{c \log n} \\ &\in \exp(-\Omega(\log^2 n)). \end{aligned}$$

Now taking the union bound over all  $v$  implies the claim.  $\square$

Similarly to  $\mathcal{I}_{t_0}(v)$ , given  $\mathcal{J}_{t_0}(v)$ , we can determine the state of  $v$  at time step  $t_0$ . We say that a multigraph  $\mathcal{J}(v)$  is a *leader generating interaction pattern for node  $v$*  if  $\mathcal{J}_{t_0}(v) = \mathcal{J}(v)$  implies that node  $v$  is a leader at step  $t_0$ . We now show that we can unfold such multigraphs into trees, without increasing their size too much.

**Lemma 35** *Suppose  $\mathcal{J}(v)$  is a leader generating interaction pattern for node  $v$  with  $N$  distinct nodes and  $k$  internal interactions. Then there exists a leader generating interaction pattern  $\mathcal{K}(v)$  for node  $v$  that has  $k - 1$  internal interactions and has size at most  $2N$ .*

**Proof** Consider an internal interaction between nodes  $u$  and  $w$  in  $\mathcal{J}(v)$  at time  $r$ , such that  $r$  has the smallest value. Consider all interactions that influenced nodes  $u$  and  $w$  before step  $r$ . Because the interaction between  $u$  and  $w$  is an internal interaction with the smallest timestamp, the interactions that influenced nodes  $u$  and  $w$  before step  $r$  are given by some multigraphs of influencers  $\mathcal{I}(u)$  and  $\mathcal{I}(w)$ , which are trees disjoint in edges and vertices. We now construct a new multigraph  $\mathcal{K}$  as follows:

- First, remove the interaction between  $u$  and  $w$  at step  $r$  from  $\mathcal{J}(v)$ . Additionally, for every interaction with timestamp larger than  $r$  (i.e., interaction that happened after  $u$  and  $w$  interacted), shift the value of its timestamp ahead by  $2r + 1$ . This way, every interaction in the current multigraph has timestamp either less than  $r$  or greater than  $3r + 1$ .
- Second, create isomorphic and disjoint copies  $\mathcal{I}(u')$  and  $\mathcal{I}(w')$  of the multigraphs of influencers  $\mathcal{I}(u)$  and  $\mathcal{I}(w)$ , respectively. Add these interactions (and the new nodes that take part in these interactions) to the multigraph  $\mathcal{K}$ . Shift all timestamps of  $\mathcal{I}(u')$  ahead by  $r$  and shift all timestamps of  $\mathcal{I}(w')$  ahead by  $2r$ . That way, timestamps of all interactions given by  $\mathcal{I}(u')$  is between  $r$  and  $2r - 1$ , timestamps of all interactions given by  $\mathcal{I}(w')$  are between  $2r$  and  $3r - 1$ .
- Finally, add the interactions  $(u, w')$  and  $(u', w)$  at time step  $3r$  and  $3r + 1$ .

See Figure 1 for an illustration of the above construction. First, note that interactions in the graph  $\mathcal{K}$  have distinct timestamps. Additionally, note that the new interactions at steps  $3r$  and  $3r + 1$  will update the states of nodes  $u$  and  $w$  the same way in  $\mathcal{K}$ , as the interaction  $(u, w)$  did previously, because  $u'$  and  $w'$  are influenced by isomorphic interaction sequences as  $u$  and  $w$  were in  $\mathcal{J}(v)$ , and since the nodes are anonymous in the population model and have no inputs in the leader election problem. In particular, this means that node  $v$  will have the same state after the system has

executed interactions in  $\mathcal{K}$  as it would have when executing interactions from  $\mathcal{J}(v)$ . Thus,  $\mathcal{K}$  is also a leader generating pattern for  $v$ .

Finally, observe that  $\mathcal{K}$  has at most  $2N$  vertices: by adding  $\mathcal{I}(u')$  and  $\mathcal{I}(w')$  to  $\mathcal{K}$  we at most doubled the number of vertices in  $\mathcal{J}(v)$ , as  $\mathcal{J}(v)$  contained its copies of  $\mathcal{I}(u)$  and  $\mathcal{I}(w)$  initially. Moreover,  $\mathcal{K}$  has at most  $k - 1$  internal interactions, as we removed at least one internal interaction of  $\mathcal{J}(v)$  without adding any new ones.  $\square$

We are now ready to prove the lower bound.

**Theorem 8** *Let  $0 < \lambda < 1$  and  $0 < \phi < 1$  be constants. If  $G$  has minimum degree  $\delta \geq \lambda n^\phi$  and  $m \geq \lambda n^2$  edges, then any stable leader election protocol requires  $\Omega(n \log n)$  expected steps to stabilize on  $G$ .*

**Proof** By Lemma 33, there exist constants  $\varepsilon > 0$  and  $c_0 > 0$  such that for any  $t \leq c_0 n \log n$ , the set  $S(t)$  contains any tree of size  $n^{\varepsilon+c_0}$  with high probability. Moreover, let  $c_1 > 0$  be the constant obtained from Lemma 34 with this  $\varepsilon$ . Let  $c = \min\{c_0, c_1\}$ . Let  $\mathcal{P}$  be any a leader election protocol on  $G$ . Consider the configuration  $x$  at any fixed time step  $t \leq cn \log n$ . Note that for such  $t$ , the statements of Lemma 34 and Lemma 33 hold with high probability; we condition the rest of the proof on the event that both statements hold. We show that in such case  $x$  is not a stable configuration.

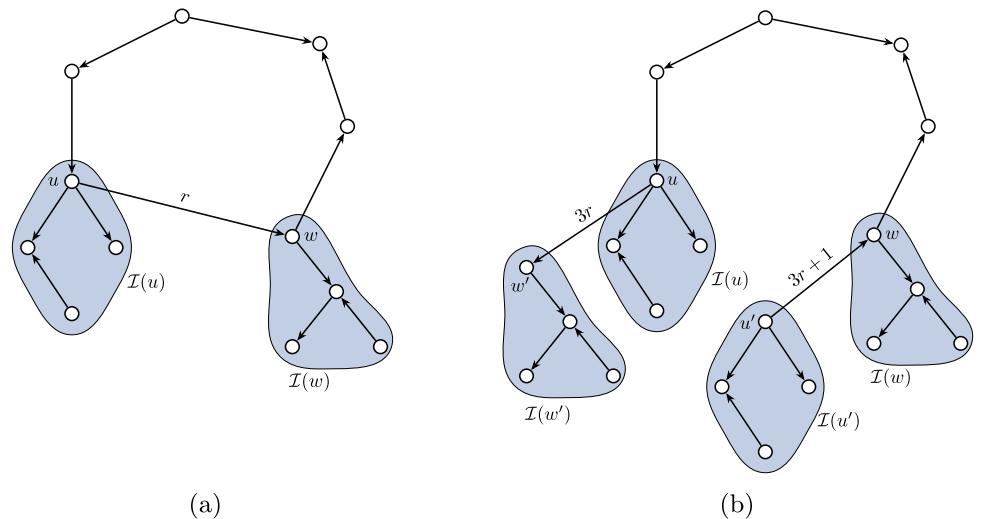
Suppose  $x$  is stable and  $v$  is the elected leader. Now  $\mathcal{J}(v) = \mathcal{J}_t(v)$  is a leader generating pattern. By Lemma 34,  $\mathcal{J}(v)$  has at most  $c \log n$  internal interactions and size at most  $n^\varepsilon$  with high probability. By applying Lemma 35 repeatedly for  $c \log n$  times, we can obtain a new pattern  $\mathcal{J}'(v)$  with size  $n^{\varepsilon+c}$  without any internal interactions. That is, the edges in  $\mathcal{J}'(v)$  (when the orientation of the edges is ignored) form a tree  $T$  of size at most  $n^{\varepsilon+c}$ ; note that absence of internal interactions also means absence of edges of multiplicity higher than 1.

By Lemma 33, the subgraph induced by  $S(t)$  contains a subgraph isomorphic to  $T$ . In particular, this means that we can construct a leader generating interaction pattern  $\mathcal{J}(w)$  isomorphic to  $\mathcal{J}(v)$  for some node  $w \in S(t)$  using only nodes of  $S(t)$ . This means that there is an execution, where node  $w \in S(t)$  will be elected as a leader. Hence,  $x$  is not a stable configuration with high probability. Thus, the stabilization time  $T_{\mathcal{P}}$  of the protocol  $\mathcal{P}$  satisfies

$$\mathbf{E}[T_{\mathcal{P}}] = \sum_{i=0}^{\infty} \Pr[T_{\mathcal{P}} > i] \geq \sum_{i=0}^t \Pr[T_{\mathcal{P}} > i] \geq Ct$$

for some constant  $C > 0$ . Thus,  $\mathbf{E}[T_{\mathcal{P}}] \in \Omega(n \log n)$ .  $\square$

**Fig. 1** Illustration of the proof strategy of Lemma 35. (a) Multigraph  $\mathcal{J}(v)$  with an internal interaction between  $u$  and  $w$ . (b) Multigraph  $\mathcal{K}$  with one internal interaction less.



## 7 Lower bound for constant-state protocols

In this section, we consider average-case lower bounds for constant-state protocols that guarantee stabilization in finite expected time on any connected graph. In the clique, the classic approach has been to utilize the so-called *surgery technique* of Doty and Soloveichik [9], later extended by Alistarh et al. [11] to show lower bounds for super-constant state protocols. Roughly, surgeries consist of carefully “stitching together” transition sequences, in order to completely eliminate states whose count decreases too fast (e.g., the leader state), thus resulting in incorrect executions (e.g., executions without a leader).

When moving beyond cliques, applying surgeries is difficult. The key challenge is that in addition to keeping track of the counts of states, we also need to control the *spatial distribution* of generated states in order to determine if a given configuration is stable. For example, if we know that an interaction  $(a, b) \rightarrow (c, d)$  produces a leader, then in the case of a clique it suffices to check if states  $a$  and  $b$  are present in the overall population to determine if this interaction can produce a new leader. However, in the case of general interaction graphs, the rule  $(a, b) \rightarrow (c, d)$  can only produce a leader if some nodes with states  $a$  and  $b$  in the configuration are *adjacent*.

We circumvent this obstacle by considering a *random graph setting*, where the interaction graph itself is *probabilistic*. Instead of showing a lower bound for a given graph, we give a lower bound that holds in *most* graphs, where “most” is interpreted as having graphs coming from a certain probability distribution. We will focus on the Erdős–Rényi random graph model  $G_{n,p}$ . As we are only interested in connected graphs  $G$ , we adopt the convention that stabilization time  $T_{\mathcal{P}}(G) = \infty$  if  $G$  is disconnected. Let  $G_{n,p}^c$  be the distribution of an Erdős–Rényi random graph conditioned on the

sampled graph being connected. Our main result is then the following.

**Theorem 9** *Let  $\mathcal{P}$  be a constant-state protocol that stabilizes in finite expected time on any connected graph. Fix a constant  $p > 0$ , and let  $G \sim G_{n,p}^c$ . Then the expected stabilization time of  $\mathcal{P}$  on  $G$  satisfies  $\mathbf{E}[T_{\mathcal{P}}(G)] \in \Omega(n^2)$ .*

In the above, the state transition function of the protocol must be independent from the realized graph. Furthermore, randomness is both over the draw of the random graph and the execution of the protocol. In addition, we condition on  $G$  being connected in order for our lower bound to be meaningful: indeed, if  $G$  is disconnected with positive probability, then any protocol has infinite expected stabilization time.

This result generalizes the lower bound of Doty and Soloveichik [9] from cliques to dense random graphs. As such, we follow a similar approach, but provide new ideas to deal with the structure of the interaction graph. For the remainder of the proof, we assume that for infinitely many  $n$  the protocol  $\mathcal{P}$  stabilizes on a random graph  $G \sim G_{n,p}^c$  in  $o(n^2)$  expected steps and show that this leads to a contradiction. It will be useful to have a tail bound on stabilization time of  $\mathcal{P}$ . Specifically, we have the following simple lemma.

**Lemma 36** *Suppose  $\mathcal{P}$  stabilizes on a random graph  $G \sim G_{n,p}^c$  in  $o(n^2)$  expected steps. Then there exists an (integer) time step  $t \in o(n^2)$  such that  $\mathcal{P}$  stabilizes on  $G \sim G_{n,p}$  in  $t$  steps with probability  $1 - o(1)$ .*

**Proof** By our assumption, we have  $\mathbf{E}[T_{\mathcal{P}}(G)] \in o(n^2)$  as  $G \sim G_{n,p}^c$ . Consider the time step

$$t = \left\lceil \sqrt{n^2 \mathbf{E}[T_{\mathcal{P}}(G)]} \right\rceil.$$

Note that  $t \in o(n^2)$  and

$$\Pr[T_{\mathcal{P}}(G) > t] \leq \sqrt{\frac{\mathbb{E}[T_{\mathcal{P}}]}{n^2}} \in o(1)$$

by Markov's inequality, which gives control on the tail of  $T_{\mathcal{P}}$ . It is also well known that  $G \sim G_{n,p}$  with  $p \in \Theta(1)$  is connected with probability  $1 - o(1)$  (e.g., see [49, Exercise 2.14]). Then, as  $G \sim G_{n,p}$ , we have by the law of total probability,

$$\begin{aligned} \Pr[T_{\mathcal{P}}(G) > t] &= \Pr[T_{\mathcal{P}}(G) > t \mid G \text{ is connected}] \Pr[G \text{ is connected}] \\ &\quad + \Pr[T_{\mathcal{P}}(G) > t \mid G \text{ is disconnected}] \Pr[G \text{ is disconnected}] \\ &\leq \Pr[T_{\mathcal{P}}(G) > t \mid G \text{ is connected}] + \Pr[G \text{ is disconnected}] \\ &\in o(1) + o(1) = o(1), \end{aligned}$$

as desired.  $\square$

Note that we additionally switched from  $G_{n,p}^c$  to  $G_{n,p}$  in the conclusion of the above lemma, since sampling from  $G_{n,p}$  is more convenient in the following.

First, we show that any protocol starting from a uniform initial configuration, passes through a “fully dense” configuration with very high probability on a sufficiently-dense Erdős–Rényi random graph. Second, we show that if the protocol stabilizes too fast, then there exist reachable configurations with many states in low count. Finally, we use surgeries to show that such a protocol must fail even on the clique.

## 7.1 First step: Reaching fully dense configurations

Fix the protocol  $\mathcal{P}$  as in Theorem 9. Let  $\Lambda$  be the set of states of the protocol. Let  $s_{\text{init}}$  be the initial state in which all nodes start. Without loss of generality, we assume that every state in  $\Lambda$  is producible by some schedule.

For  $\alpha > 0$ , a configuration  $x$  is said to be  $\alpha$ -dense with respect to a set  $\Lambda' \subseteq \Lambda$  if every state in  $\Lambda'$  is present in count at least  $\alpha n$ . A configuration is *fully  $\alpha$ -dense* with respect to  $\Lambda'$  if additionally all other states have count zero. We now show that the protocol reaches a fully  $\alpha$ -dense configuration with respect to  $\Lambda$  with probability  $1 - \exp(-\Omega(n))$  on a random Erdős–Rényi graph. For any two sets  $S, T \subseteq V(G)$ , we write  $e(S, T)$  for the number edges between nodes in  $S$  and  $T$ .

**Lemma 37** *Let  $p, c > 0$  be constants and  $G \sim G_{n,p}$ . Then, the following events hold with probability  $1 - \exp(-\Omega(n))$  for all disjoint  $S, T \subseteq V(G)$  with  $|S|, |T| > cn$  simultaneously*

- $p|S||T|/2 \leq e(S, T) \leq 3p|S||T|/2$  and

- $p|S|^2/4 \leq e(S, S) \leq 3p|S|^2/4$ .

**Proof** Note that in an Erdős–Rényi random graph, the number  $e(S, T)$  of edges between  $S$  and  $T$  follows the binomial distribution  $\text{Bin}(|S||T|, p)$  with mean  $\mu = p|S||T| \in \Theta(n^2)$ . Using standard Chernoff bounds (Lemma 2), we get that

$$\begin{aligned} \Pr\left[|e(S, T) - \mu| < \frac{\mu}{2}\right] \\ > 1 - 2 \cdot \exp\left(-\frac{\mu}{12}\right) \in 1 - \exp\left(-\Omega\left(n^2\right)\right). \end{aligned}$$

The first claim follows by applying the union bound over at most  $4^n$  pairs of sets. Observe that  $e(S, S)$  follows a binomial distribution of  $\text{Bin}(|S|(|S| - 1)/2, p)$  with mean  $\mu' = p|S|(|S| - 1)/2 \in \Theta(n^2)$ . Therefore, the second claim also follows by an analogous argument.  $\square$

**Lemma 38** *Let  $p > 0$  be a constant and  $G \sim G_{n,p}$ . Then there exists a constant  $\alpha = \alpha(p, |\Lambda|)$  such that an execution of  $\mathcal{P}$  on  $G$  reaches a fully  $\alpha$ -dense configuration with respect to  $\Lambda$  in  $O(n)$  steps with probability  $1 - \exp(-\Omega(n))$ .*

**Proof** Suppose all nodes start in the initial state  $s_{\text{init}}$ . Without loss of generality, assume that every state in  $\Lambda$  is producible by some transition sequence from the initial configuration. Then there exists a sequence of sets  $\{s_{\text{init}}\} = \Lambda_1 \subsetneq \dots \subsetneq \Lambda_{|\Lambda|} = \Lambda$  such that the state  $b_{i+1} \in \Lambda_{i+1} \setminus \Lambda_i$  is producible using some interaction between nodes with states from  $\Lambda_i$ . Let  $f(1) = p/4$  and for  $i \geq 1$  define

$$f(i+1) = f^2(i) \cdot \frac{p}{960}.$$

Note that  $f$  is decreasing and  $f(i+1) \leq f(i)/2$  for every  $i \geq 1$ . By Lemma 37 applied with the constant  $c = f(|\Lambda| + 1)$ , all of the following hold with probability  $1 - \exp(-\Omega(n))$ :

1. The number of edges in  $G$  satisfies  $pn^2/4 \leq m \leq 3pn^2/4$ .
2. For any disjoint sets  $U, U' \subseteq V$  of size at least  $nf(i)$ , we have that  $e(U, U') \geq |U||U'|p/2$  for  $1 \leq i \leq |\Lambda| + 1$ .
3. For any set  $U \subseteq V$  of size at least  $nf(i)$ , we have  $e(U, U) \geq |U|^2p/4$  for  $1 \leq i \leq |\Lambda| + 1$ .

We say that an event holds with very high probability if it holds with probability  $1 - \exp(-\Omega(n))$ . We now prove by induction on  $i$  that the protocol reaches an  $f(i)$ -dense configuration with respect to  $\Lambda_i$  with very high probability. This implies that we reach a  $f(|\Lambda|)$ -dense configuration with very high probability. The base case  $i = 1$  of the induction is vacuous, as the initial configuration is 1-dense with respect to  $\Lambda_1 = \{s_{\text{init}}\}$ , and hence, also  $f(1)$ -dense with respect to  $\Lambda_1$ . We say that a state  $s$  has density  $\rho$  in a configuration if there are  $\rho n$  nodes in state  $s$ .

For the inductive step, suppose we have reached a configuration that is  $f(i)$ -dense with respect to  $\Lambda_i$  for some  $1 \leq i < |\Lambda|$ . Suppose this happens at step  $t$ . Let  $s \in \Lambda_i$  and define  $N = np/32$ . By the induction assumption, the density of  $s$  is at least  $f(i)$  at step  $t$ .

First, we show that the density of  $s$  remains at least  $f(i)/2 \geq f(i+1)$  for the next  $N = np$  steps with very high probability. Suppose that the density of  $s$  goes below  $f(i)/2$  before step  $t + N$  and let  $t' < t + N$  be the last time when the density is at least  $f(i)$ . Note that if the count of  $s$  decreases, it can decrease by at most two. The claim follows if we can show that with very high probability the count of  $s$  decreases at most  $nf(i)/4$  times during the interval  $I = \{t', t' + 1, \dots, t + N\}$ , which would be a contradiction. During this interval, the density of  $s$  is at most  $f(i)$  and the probability of the density decreasing in a single step is at most the probability that a stochastic scheduler samples an edge incident with a node in state  $s$ . Since every node in state  $s$  has degree at most  $n$ , the probability is at most

$$r(i) = \frac{f(i)n^2}{m} \leq \frac{4f(i)}{p} \leq 1,$$

where the bound on  $m$  follows from (1) and the fact that  $f(i) \leq p/4$ . The number  $X(s)$  of times the density of  $s$  decreases during the interval  $I$  is stochastically dominated by the binomial random variable  $Y \sim \text{Bin}(N, r(i))$  with mean

$$\mathbf{E}[Y] = Nr(i) = \frac{npr(i)}{32} \leq \frac{np}{32} \cdot \frac{4f(i)}{p} = \frac{nf(i)}{8}.$$

Using Chernoff bounds (Lemma 2) applied with  $\lambda = 1$ , we get that

$$\begin{aligned} \Pr\left[X(s) \geq \frac{nf(i)}{4}\right] &\leq \Pr\left[Y \geq \frac{nf(i)}{4}\right] \\ &\leq \Pr[Y \geq 2 \cdot \mathbf{E}[Y]] \\ &\leq \exp(-\mathbf{E}[Y] \cdot 1/3) \in \exp(-\Omega(n)). \end{aligned}$$

By union bound, the density of every  $s \in \Lambda_i$  remains at least  $f(i)/2 \geq f(i+1)$  until step  $t + N + 1$  with very high probability.

To complete the induction, we show that the density of  $b_{i+1} \in \Lambda_{i+1} \setminus \Lambda_i$  reaches  $f(i+1)$  before step  $t + N$  with very high probability. Suppose that the density of  $b_{i+1}$  remains less than  $f(i+1)$  until step  $t + N$ . Similarly, as in the argument above, the probability that the count of  $b_{i+1}$  decreases given that its density is at most  $f(i+1)$  is at most

$$r(i+1) = \frac{f(i+1)n^2}{m} \leq \frac{4f(i+1)}{p} \leq 1.$$

Let  $s, s' \in \Lambda_i$  be a pair of states whose interaction generates the state  $b_{i+1}$  (such states must exist by definition of  $b_{i+1}$ ). The probability that the density of  $b_{i+1}$  increases is the probability that some pair of nodes  $(v, u)$  is sampled by the stochastic scheduler, where  $v$  is in state  $s$  and  $u$  is in state  $s'$ . Above, we showed that the density of  $s$  and  $s'$  is at least  $f(i)/2$ , so the probability that the density of  $b_{i+1}$  increases is at least

$$\begin{aligned} q(i+1) &= \frac{(nf(i)/2)^2 p}{2 \cdot 2m} \geq \frac{(nf(i))^2 p}{16} \cdot \frac{4}{3pn^2} \\ &= \frac{f^2(i)}{12} \geq \frac{960f(i+1)}{12p} \\ &= \frac{80f(i+1)}{p}, \end{aligned}$$

where we have used properties (1)–(3), the fact that  $f(i)/2 \geq f(i+1) \geq f(|\Lambda|+1)$ , and the definition  $f(i+1) = f^2(i)p/960$ . Suppose the density of  $b_{i+1}$  is at most  $f(i+1)$  during the interval  $t, \dots, t+N$ . We will show that this implies that with very high probability by time step  $t + N$  the number of steps during which the count of  $b_{i+1}$  increases is at least  $2f(i+1)$ , and the number of decrease events is at most  $f(i+1)/2$ . Since every step that decreases the count of  $b_{i+1}$  can decrease its count by at most two, this implies that the density is at least  $f(i+1)$  by time  $t + N$ .

Given that the density of state  $b_{i+1}$  remains at most  $f(i+1)$  during the interval  $t, \dots, t+N$ , the number of steps that increase the density stochastically dominates the binomial random variable  $Y \sim \text{Bin}(N, q(i+1))$ . The number of steps that decrease the count of  $b_{i+1}$  is in turn stochastically dominated by a binomial random variable  $Z \sim \text{Bin}(N, r(i+1))$ . Now using standard Chernoff bounds (Lemma 2), we have that

$$\begin{aligned} \Pr[Y \leq \mathbf{E}[Y]/2] &\in \exp(-\Omega(n)) \quad \text{and} \\ \Pr[Z \geq 2 \cdot \mathbf{E}[Z]] &\in \exp(-\Omega(n)). \end{aligned}$$

That is, with very high probability the number of increases is at least  $\mathbf{E}[Y]/2$  and the number of decreases at most  $2 \cdot \mathbf{E}[Z]$ . From the above bounds on  $r(i+1)$  and  $q(i+1)$ , we get that

$$\begin{aligned} \mathbf{E}[Y]/2 - 2 \cdot \mathbf{E}[Z] &= \frac{Nq(i+1)}{2} - 2Nr(i+1) \\ &\geq N \cdot \left( \frac{80f(i+1)}{2p} - \frac{8f(i+1)}{p} \right) \\ &= nf(i+1), \end{aligned}$$

since  $N = np/32$ . Therefore, the density of  $b_{i+1}$  must be at least  $f(i+1)$  by step  $t + N$  with very high probability.  $\square$

In the proof above, the only information we use about the interaction graph is the information provided by Lemma 37.

We note that Lemma 38 also holds for deterministic graphs having similar connectivity properties as guaranteed by Lemma 37. In particular, any dense graph with small spectral gap would also produce fully dense configurations with high probability.

## 7.2 Second step: Controlling for low count states

A set  $S \subseteq \Lambda$  is *leader generating* if  $S$  either contains a leader state (i.e., a state in which node's output is leader), or given enough nodes in each state of  $S$ , there exists a finite sequence  $\rho = (e_1, \dots, e_t)$  of interactions such that these nodes generate a leader on the clique. Alistarh et al. [11, Lemma A.7] showed that for any leader generating set  $S$  we need at most  $2^{|\Lambda|}$  nodes in each state to generate a leader on the clique, i.e., a clique of size  $|S| \cdot 2^{|\Lambda|}$  suffices to generate a new leader state. Say that the state  $s$  is *present in low count* in the configuration  $x$  if its count is less than  $2^{|\Lambda|}$  in  $x$ . Otherwise, we will say that it is *present in high count*.

The goal of this subsection is to show that, if the protocol stabilizes in  $t \in o(n^2)$  steps with high probability, then, with at least constant probability, after  $t$  steps no leader generating set has all states in high counts. More formally, the goal of this subsection is to prove the following lemma.

**Lemma 39** *Let  $t \in \mathbb{N}$  be such that, as  $G \sim G_{n,p}$ , we have*

$$\Pr[T_{\mathcal{P}}(G) > t] \in o(1).$$

*Let  $x_t$  be a configuration reached by the protocol after  $t$  steps. Let  $\xi(x_t)$  be a set of states in low count in  $x_t$ . Then, we have that  $\xi(x_t) \cap S \neq \emptyset$  for all leader generating sets  $S$  with probability  $\Theta(1)$  bounded away from zero.*

The utility of the above lemma is as follows. Having control on the counts of states in leader generating sets will allow us to apply a variant of the “surgery technique” of [9] and construct an execution with no leader and no possibility to elect a leader, thus leading to a contraction.

We now elucidate why the proof of the above result is non-trivial. Consider the following informal attempt at proving Lemma 39. Suppose, for the sake of contradiction, that with probability  $1 - o(1)$ , there exist a leader generating set  $S \subseteq \Lambda$  for which  $x_t$  contains all states of  $S$  in high counts. Take  $2^{|\Lambda|}$  nodes in  $x_t$  of each state in  $S$ , and let  $U$  be the resulting set. By definition of a leader generating set and using once again the result of Alistarh et al. [11, Lemma A.7], we know that if nodes in  $U$  form a clique, then a new leader can be generated, and thus  $x_t$  cannot be stable. If nodes of  $U$  form a clique with at least constant probability, then the probability that  $x_t$  is not a stable configuration is at least constant as well. Thus, to achieve a contradiction, it remains to lower bound the probability of nodes in  $U$  forming a clique. This might initially seem like a trivial task: the cardinality of  $U$  is upper

bounded by  $|\Lambda|2^{|\Lambda|} \in O(1)$ , and the probability that each edge between vertices of  $U$  is present in  $G$  is  $p \in \Theta(1)$ , i.e., the resulting probability should be  $p^{\binom{|U|}{2}} \in \Theta(1)$ . However, recall that our choice of nodes  $U$  depends on the configuration  $x_t$ , and thus, on the realization of the graph  $G$ . Therefore, computing the probability of  $U$  forming a clique is a highly non-trivial task, because of its dependence with the execution of the protocol.

This prompts us to bound the correlations between realization of  $G$  and execution of the protocol. Intuitively, the protocol only has a limited view of the graph  $G$ , e.g., it can only see the edges of  $G$  that were activated as part of the stochastic schedule. We will now formalize this intuition. Let  $A = \{(u, v) : u, v \in V, u \neq v\}$ . For any sequence  $\rho = (e_1, \dots, e_t) \in A^t$  of pairs of nodes, let  $\text{Exec}(\rho)$  be the event that sequence  $\rho$  was sampled by the scheduler in the first  $t$  steps, i.e.,  $\rho$  is a prefix of the stochastic schedule. Slightly abusing the notation, we also let  $|\rho|$  denote the number of distinct edges in  $\rho$ .

In the following, we want to consider all graphs  $G \sim G_{n,p}$  in which  $\rho$  could be executed and infer something about the presence of edges in such graphs. However, we cannot simply state that  $e \in E(G)$  with probability  $p$ , because the fact that  $\rho$  was sampled by the scheduler introduces bias into the random graph  $G \sim G_{n,p}$ , i.e., the events  $e \in E(G)$  and  $\text{Exec}(\rho)$  are correlated. The next lemma shows that this correlation is small, as long as  $|\rho|$  is sufficiently large.

**Lemma 40** *Let  $t \geq 1$  be a positive integer and  $F \subseteq A$ . Let  $\rho$  be a fixed sequence of edges of length  $t$ . Then the random graph  $G = (V, E) \sim G_{n,p}$  satisfies*

$$\Pr[F \subseteq E \mid \text{Exec}(\rho)] \geq \left[ p \left( 1 - \frac{1}{|\rho|} \right)^t \right]^{|F|}.$$

**Proof** The event  $\text{Exec}(\rho)$  implies that pairs in  $\rho$  are contained in  $E$ , that is, for every  $(v, u) \in \rho$  we have  $\{v, u\} \in E$ . In other words, all necessary edges to perform the interactions in  $\rho$  are present in the random graph. We write  $E(\rho)$  for the edges in  $\rho$  and use the shorthand  $\rho \subseteq E$  to denote the event that all edges in  $E(\rho)$  are contained in  $E$ . Then by Bayes' law, we get

$$\begin{aligned} \Pr[F \subseteq E \mid \text{Exec}(\rho)] &= \Pr[F \subseteq E \mid \text{Exec}(\rho) \wedge \rho \subseteq E] \\ &= \frac{\Pr[\text{Exec}(\rho) \mid F \subseteq E \wedge \rho \subseteq E]}{\Pr[\text{Exec}(\rho) \mid \rho \subseteq E]} \\ &\quad \cdot \Pr[F \subseteq E \mid \rho \subseteq E] \\ &\geq \frac{\Pr[\text{Exec}(\rho) \mid F \subseteq E \wedge \rho \subseteq E]}{\Pr[\text{Exec}(\rho) \mid \rho \subseteq E]} \\ &\quad \cdot \Pr[F \subseteq E] \\ &= \frac{\Pr[\text{Exec}(\rho) \mid F \subseteq E \wedge \rho \subseteq E]}{\Pr[\text{Exec}(\rho) \mid \rho \subseteq E]} \cdot p^{|F|}, \end{aligned}$$

where for the inequality we used the fact that

$$\begin{aligned}\Pr[F \subset E \wedge \rho \in E] &= \Pr[(F \cup E(\rho)) \subset E] \\ &\geq \Pr[F \subset E] \cdot \Pr[\rho \subset E].\end{aligned}$$

Let  $F = \{f_1, \dots, f_h\}$ . We use  $Q_0$  to denote the event that  $\rho \subseteq E$  happens. For  $i \geq 1$ , define the events  $Q_i = Q_{i-1} \wedge f_i \in E$  and  $Q_i^* = Q_{i-1} \wedge f_i \notin E$ . Next, we show that the following holds for all  $i \geq 1$ :

$$\frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_{i-1}]} \geq \left(1 - \frac{1}{N}\right).$$

The statement is trivial if  $f_i \in E(\rho)$ , as in that case  $Q_i = Q_{i-1}$ . Suppose then that  $f_i \notin E(\rho)$ . We now want to lower bound the ratio

$$\begin{aligned}R_i &= \frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_i] \cdot \Pr[f_i \in E \mid Q_{i-1}] + \Pr[\text{Exec}(\rho) \mid Q_i^*] \cdot \Pr[f_i \notin E \mid Q_{i-1}]} \\ &= \frac{1}{\Pr[f_i \in E \mid Q_{i-1}] + \frac{\Pr[\text{Exec}(\rho) \mid Q_i^*]}{\Pr[\text{Exec}(\rho) \mid Q_i]} \cdot \Pr[f_i \notin E \mid Q_{i-1}]} \\ &\geq \frac{1}{\Pr[f_i \in E \mid Q_{i-1}] + (1 - 1/|\rho|)^{-t} \cdot \Pr[f_i \notin E \mid Q_{i-1}]} \\ &\geq \frac{1}{(1 - 1/|\rho|)^{-t} \cdot (\Pr[f_i \in E \mid Q_{i-1}] + \Pr[f_i \notin E \mid Q_{i-1}])} = \left(1 - \frac{1}{|\rho|}\right)^t.\end{aligned}$$

$$\frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_i^*]}.$$

Note that if we know  $\rho \subseteq E$  holds, then  $\Pr[\text{Exec}(\rho) \mid G] = 1/(2m)^t$ , where  $m$  is the number of edges in  $E$ . It follows then that

$$\Pr[\text{Exec}(\rho) \mid Q_i] = \mathbf{E}[(2m)^{-t} \mid Q_i].$$

Similarly,  $\Pr[\text{Exec}(\rho) \mid Q_i^*] = \mathbf{E}[(2m)^{-t} \mid Q_i^*]$ . That is, we have

$$\frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_i^*]} = \frac{\mathbf{E}[(2m)^{-t} \mid Q_i]}{\mathbf{E}[(2m)^{-t} \mid Q_i^*]} = \frac{\mathbf{E}[m^{-t} \mid Q_i]}{\mathbf{E}[m^{-t} \mid Q_i^*]}.$$

Let  $N \geq |\rho|$  be the number of distinct edges among  $E(\rho) \cup \{f_1, \dots, f_{i-1}\}$ . Note that, conditioned on  $Q_i$ ,  $m$  follows the distribution of a shifted binomial random variable  $\text{Bin}\left(\frac{n(n-1)}{2} - N - 1, p\right) + N + 1$ . Similarly, conditioned on  $Q_i^*$ ,  $m$  follows the distribution of  $\text{Bin}\left(\frac{n(n-1)}{2} - N - 1, p\right) + N$ . Let  $X \sim \text{Bin}\left(\frac{n(n-1)}{2} - N - 1, p\right)$ . Then,

$$\begin{aligned}\frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_i^*]} &= \frac{\mathbf{E}[(X + N + 1)^{-t}]}{\mathbf{E}[(X + N)^{-t}]} \\ &= \frac{\sum_{x \geq 0} \Pr[X = x] (x + N + 1)^{-t}}{\sum_{x \geq 0} \Pr[X = x] (x + N)^{-t}} \\ &\geq \left(1 - \frac{1}{N}\right)^t \geq \left(1 - \frac{1}{|\rho|}\right)^t,\end{aligned}$$

where the second inequality follows since for any  $x \geq 0$  the ratio  $\frac{(x+N+1)^{-t}}{(x+N)^{-t}}$  is lower bounded by  $(1 - 1/N)^t$ . Next, consider the ratio

$$R_i = \frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_{i-1}]}.$$

Observe that

Therefore,

$$\begin{aligned}\frac{\Pr[\text{Exec}(\rho) \mid F \subseteq E \wedge \rho \subseteq E]}{\Pr[\text{Exec}(\rho) \mid \rho \subseteq E]} \\ = \prod_{i=1}^h R_i = \prod_{i=1}^h \frac{\Pr[\text{Exec}(\rho) \mid Q_i]}{\Pr[\text{Exec}(\rho) \mid Q_{i-1}]} \geq \left(1 - \frac{1}{|\rho|}\right)^{t|F|},\end{aligned}$$

which concludes the proof.  $\square$

The above lemma can be seen as an adaptation of the ‘‘principle of deferred decisions’’ in our setup. Indeed, the lemma shows that we can defer the sampling of a part of  $G$  of constant size until a later point in the execution of  $\mathcal{P}$ , while paying a small price in the bias of edges, provided  $(1 - 1/|\rho|)^t$  is not too small. Below, we show that  $(1 - 1/|\rho|)^t$  is typically lower bounded by a constant, when  $\rho$  is sampled by a stochastic schedule. Specifically, let  $\sigma_t$  be the length- $t$  prefix of the stochastic schedule. Then, we prove that  $|\sigma_t|$  is of the same order as  $t$  with high probability.

**Lemma 41** *Suppose  $t \in \mathbb{N}$  is such that  $t \in o(n^2)$ . Then we have  $|\sigma_t| \geq \lfloor t/2 \rfloor$  with probability  $1 - \exp(-\Omega(t)) - \exp(-\Omega(n))$ .*

**Proof** Let  $m$  be the number of edges in graph  $G$ . Note that, since  $t \in o(n^2)$  and  $m \in \Omega(n^2)$  with probability  $1 - \exp(-\Omega(n))$  by Lemma 37, we can ensure  $m \geq 2t$  with probability  $1 - \exp(-\Omega(n))$ . In the rest of the proof, condition on this event (i.e., assume  $m \geq 2t$  is given).

The remainder of the proof is similar to the coupon collector argument. Let  $s$  be arbitrary, and suppose  $|\sigma_s| = k - 1$ . Note that the number of edges that need to be sampled to reach  $k$  distinct edges follows a geometric distribution  $\text{Geom}\left(\frac{m-k+1}{m}\right)$ . For each  $1 \leq k \leq m$ , let  $X_k \sim \text{Geom}\left(\frac{m-k+1}{m}\right)$ . Then, we have

$$\begin{aligned} \Pr\left[|\sigma_t| \geq \left\lfloor \frac{t}{2} \right\rfloor\right] &= \Pr\left[\sum_{i=1}^{\lfloor t/2 \rfloor} X_i \leq t\right] \\ &= 1 - \Pr\left[\sum_{i=1}^{\lfloor t/2 \rfloor} X_i > t\right] \\ &\geq 1 - \Pr\left[\sum_{i=1}^{\lfloor t/2 \rfloor} Y_i > t\right], \end{aligned}$$

where  $Y_i \sim \text{Geom}\left(\frac{m-t/2}{m}\right)$ , and the last line holds since success probabilities of  $X_i$ 's are lower bounded by  $\frac{m-t/2}{m}$ . Since  $m \geq 2t$ , we have  $\frac{m-t/2}{m} \geq 3/4$ , and thus, by Lemma 3,

$$\begin{aligned} \Pr\left[\sum_{i=1}^{\lfloor t/2 \rfloor} Y_i > t\right] &\leq \Pr\left[\sum_{i=1}^{\lfloor t/2 \rfloor} Y_i > \frac{3}{2} \mathbb{E}\left[\sum_{i=1}^{\lfloor t/2 \rfloor} Y_i\right]\right] \\ &\in \exp(-\Omega(t)), \end{aligned}$$

which concludes the proof.  $\square$

Finally, we combine Lemma 41 and Lemma 40 to control the structure of the graph even when we condition on the configuration reached.

**Lemma 42** *Let  $t \in \mathbb{N}$  be such that  $t \in \omega(1)$  and  $t \in o(n^2)$ . Let  $x_t$  be a (random) configuration reached by the protocol after  $t$  steps. Let  $F(x_t)$  be a (random) set of edges that depends only on  $x_t$ . Suppose  $F(x_t) \in \Theta(1)$  holds almost surely. Then,*

$$\Pr[F(x_t) \subseteq E] \in \Theta(1).$$

**Proof** First, note that we have

$$\Pr[F(x_t) \subseteq E] \geq \Pr\left[F(x_t) \subseteq E \wedge |\sigma_t| \geq \left\lfloor \frac{t}{2} \right\rfloor\right]$$

Let  $C \in \Theta(1)$  be the almost sure upper bound on  $F(x_t)$ . Let  $\rho^{(1)}, \dots, \rho^{(k)}$  be all possible interaction sequences of length

$t$  that have at least  $\lfloor t/2 \rfloor$  distinct edges, and let  $x^{(1)}, \dots, x^{(k)}$  be the corresponding configurations reached (i.e.,  $x^{(i)}$  is the configuration reached when  $\rho^{(i)}$  is executed by the stochastic schedule, for  $1 \leq i \leq k$ ). For  $1 \leq i \leq k$ , let  $F^{(i)} = F(x^{(i)})$ . Then, for any  $1 \leq i \leq k$ , by Lemma 40, we have

$$\begin{aligned} \Pr\left[F^{(i)} \subseteq E \mid \text{Exec}(\rho^{(i)})\right] &\geq \left[p\left(1 - \frac{1}{\lfloor t/2 \rfloor}\right)^t\right]^{|F^{(i)}|} \\ &\geq \left[p\left(1 - \frac{1}{\lfloor t/2 \rfloor}\right)^t\right]^C, \end{aligned}$$

since  $|F^{(i)}| \leq C$  for all  $1 \leq i \leq k$ . Then, by the law of total probability, we get that

$$\begin{aligned} \Pr\left[F(x_t) \subseteq E \wedge |\sigma_t| \geq \left\lfloor \frac{t}{2} \right\rfloor\right] &= \sum_{i=1}^k \Pr\left[F^{(i)} \subseteq E \mid \text{Exec}(\rho^{(i)})\right] \cdot \Pr\left[\text{Exec}(\rho^{(i)})\right] \\ &\geq \left[p\left(1 - \frac{1}{\lfloor t/2 \rfloor}\right)^t\right]^C \cdot \sum_{i=1}^k \Pr\left[\text{Exec}(\rho^{(i)})\right] \\ &= \left[p\left(1 - \frac{1}{\lfloor t/2 \rfloor}\right)^t\right]^C \cdot \Pr\left[|\sigma_t| \geq \left\lfloor \frac{t}{2} \right\rfloor\right] \\ &\in \Theta(1) \cdot \Theta(1) = \Theta(1), \end{aligned}$$

with last transition following by Lemma 41 and since  $C \in \Theta(1)$ ,  $p \in \Theta(1)$ , and  $\left(1 - \frac{1}{\lfloor t/2 \rfloor}\right)^t \in \Theta(1)$  for  $t \geq 4$ .  $\square$

The above lemma gives us the power to overcome the challenge described in the introduction: we have control over the structure of the graph when conditioned on a given configuration. However, we only have meaningful control over a portion of the graph of constant size. Turns out, this is sufficient for us.

**Proof of Lemma 39** Suppose we can find in  $x_t$  a leader generating set  $S$  with all states  $s \in S$  being present in high count in  $x_t$ . That is, every state  $s \in S$  has count at least  $2^{|A|}$  in  $x_t$ . Pick  $2^{|A|}$  nodes of each state in configuration  $x$  and call this set  $U$ . As mentioned above, by [11, Lemma A.7], a leader can be elected from nodes in  $U$  given that all necessary edges between them are present. (If all leader generating sets in  $x_t$  have a low-count state, set  $U = \emptyset$ .) Let  $F^*$  be the set of edges between all nodes in  $U$ . Note that  $F^*$  is given by a deterministic function of  $x_t$  and the set has at most constant size, and thus, by Lemma 42

$$\Pr[U \text{ is a clique}] = \Pr[F^* \subseteq E] \in \Theta(1).$$

Let  $\text{Stable}(x)$  denote the event that  $x$  is a stable configuration and  $\text{LowCount}(x)$  the event that all leader generating sets in

$x$  have at least one state in low count. Then the above implies

$$\begin{aligned} \Pr \left[ \overline{\text{Stable}(x_t)} \right] &\geq \Pr \left[ U \text{ is a clique} \wedge \overline{\text{LowCount}(x_t)} \right] \\ &= \Pr \left[ U \text{ is a clique} \right] \\ &\quad - \Pr \left[ U \text{ is a clique} \wedge \text{LowCount}(x_t) \right] \\ &\geq \Pr \left[ U \text{ is a clique} \right] - \Pr \left[ \text{LowCount}(x_t) \right]. \end{aligned}$$

Rearranging the above, we get

$$\begin{aligned} \Pr \left[ \text{LowCount}(x_t) \right] &\geq \Pr \left[ U \text{ is a clique} \right] \\ &\quad - \Pr \left[ \overline{\text{Stable}(x_t)} \right] \in \Theta(1) - o(1) \\ &= \Theta(1), \end{aligned}$$

which concludes the proof.

### 7.3 Last step: Adapting the surgery technique

Lemma 39 allows us to adapt the surgery technique of [9] to the setting of incomplete graphs. In particular, we are able to prove a statement analogous to Lemma 4.4 in [9]. We borrow some definitions and notation from [9]. We call an interaction between states  $a$  and  $b$  a  $k$ -bottleneck if at the time of the interaction, the state counts of  $a$  and  $b$  are upper bounded by  $k$ . Also in the following, we use  $x \Rightarrow y$  to denote that the configuration  $y$  is reachable from the configuration  $x$  on a clique. Moreover, by  $x \Rightarrow_{\rho} y$  we denote that after executing interactions in  $\rho$  starting from configuration  $x$ , we end up in configuration  $y$ .

**Lemma 43** *Suppose  $\mathcal{P}$  stabilizes in  $o(n^2)$  steps with probability  $1 - o(1)$  on  $G \sim G_{n,p}$  for infinitely many  $n$ . Then there exists an infinite set  $K \subseteq \mathbb{N}$  of indices and set  $\xi \subseteq \Lambda$  of states such that for every leader generating set  $S$  we have  $\xi \cap S \neq \emptyset$  and there exist an infinite sequence of configurations  $((x^{(k)}, y^{(k)}))_{k \in K}$  along with transition sequences  $(\rho^{(k)})_{k \in K}$  such that the following holds*

1.  $x^{(k)} \Rightarrow_{\rho^{(k)}} y^{(k)}$  and  $\rho^{(k)}$  does not contain any  $k$ -bottleneck interactions,
2.  $x^{(k)}$  contains every state in count at least  $k$ , and
3.  $y^{(k)}$  contains all states from  $\xi$  in low count.

**Proof** Suppose  $n$  is such that  $\mathcal{P}$  takes  $o(n^2)$  steps to stabilize with probability  $1 - o(1)$ . We first condition on the event  $|E| > pn^2/4$ . Since this holds with probability  $1 - o(1)$  by Lemma 37, this will not affect the high-probability asymptotics of the stabilization time of the protocol. Next, we apply the union bound over Lemma 38 and Lemma 39 to conclude that with at least constant positive probability, the stochastic schedule  $\sigma$  satisfies the following properties provided that  $n$  is large enough:

- (a)  $\sigma$  passes through a fully dense configuration by time  $t$  (Lemma 38), and
- (b) in step  $t$ , the system is in a stable configuration, where every leader generating set has a state in low count (Lemma 39).

Suppose every such sequence contains a  $k$ -bottleneck interaction for some constant  $k \in \mathbb{N}$  independent of  $n$ . Note that at every time step there are at most  $|\Lambda|^2$  pairs of states with state counts at most  $k$ . Additionally, two nodes in states  $a$  and  $b$  with state count at most  $k$  interact with probability at most  $\frac{k^2}{m}$ . Then the probability of a  $k$ -bottleneck interaction occurring at an arbitrary time step is at most

$$\frac{8k^2|\Lambda|^2}{pn^2}$$

by definition of  $k$ -bottleneck transitions and since  $m \geq pn^2/4$ . Since (a) and (b) hold with at least constant positive probability, this implies that the stabilization time is of order  $\Omega(n^2)$  with probability at least  $\Omega(1)$  (similarly to [9, Observation 4.1]), contradicting our assumption. Hence, for all large enough  $n$ ,

- by property (a), we get that there exists a dense configuration  $x^{(k)}$ , and
- by property (b), we get that there exists a configuration  $y^{(k)}$  where every leader generating set has a state in low count, along with a  $k$ -bottleneck free transition sequence  $\rho^{(k)}$  such that  $x^{(k)} \Rightarrow_{\rho^{(k)}} y^{(k)}$  (as shown above).

From such  $k$ -bottleneck free transition sequences, we may form a sequence  $((x^{(k)}, y^{(k)}, \rho^{(k)}))_{k \in \mathbb{N}}$  of configurations and transition sequences which satisfy first two conditions of the lemma. Additionally, for every leader generating set,  $y^{(k)}$  has a state from it in low count. Notice however that for every  $k$  we may have a different set  $\xi$  of low count states. This can be resolved using the pigeonhole principle: since there are at most  $2^{|\Lambda|}$  possibilities for  $\xi \subseteq \Lambda$ , we can select  $K \subseteq \mathbb{N}$  and an infinite subsequence  $((x^{(k)}, y^{(k)}, \rho^{(k)}))_{k \in K}$  of  $((x^{(k)}, y^{(k)}, \rho^{(k)}))_{k \in \mathbb{N}}$  such that  $\xi$  is the same for all  $y^{(k)}$ .  $\square$

With Lemma 43, Theorem 9 now follows using standard surgery techniques [9]. Consider the execution of sequences from Lemma 43 on the clique. This setting is the same as the one considered by Doty and Soloveichik [9] with our Lemma 43 corresponding to their Lemma 4.4. The rest of the proof is now almost identical to proof by Doty and Soloveichik [9]. For the sake of completeness, we now state the necessary results needed to finish the proof of Theorem 9; we refer to [9] for details.

First, the transition ordering lemma follows from Lemma 43 (this corresponds to Lemma 4.5 from [9]):

**Lemma 44** *Let  $b_1, b_2$  be positive integers such that  $b_2 > |\Lambda|b_1$  and  $K, \xi$  and  $(x^{(k)}, y^{(k)}, \rho^{(k)})_{k \in K}$  be as in Lemma 43. For any  $k \in K$ , there is an order on  $\xi = \{d_1, \dots, d_l\}$  such that for all  $i \in \{1, \dots, l\}$  there exists a transition  $\alpha_i$ , where node in state  $d_i$  transitions to state  $o_1$  and node in state  $s_i$  transitions to state  $o_2$ , such that*

- $s_i, o_1, o_2 \notin \{d_1, \dots, d_l\}$ , and
- $\alpha_i$  occurs at least  $(b_2 - b_1|\Lambda|)/|\Lambda|^2$  times in  $\rho^{(k)}$ .

Recall that a configuration  $x$  is a map  $V \rightarrow \Lambda$ . We say two configurations  $x_1 : V_1 \rightarrow \Lambda$  and  $x_2 : V_2 \rightarrow \Lambda$  are *disjoint* if  $V_1 \cap V_2 = \emptyset$ . Then, for two disjoint configurations  $x_1 : V_1 \rightarrow \Lambda$  and  $x_2 : V_2 \rightarrow \Lambda$ , define  $x_1 + x_2$  to be a configuration on  $V_1 \cup V_2$ , where

$$(x_1 + x_2)(v) = \begin{cases} x_1(v) & \text{if } v \in V_1 \\ x_2(v) & \text{otherwise.} \end{cases}$$

With the above, we get the following lemma, which corresponds to (slightly adapted) Claims 1–3 given in [9]:

**Lemma 45** *Take  $((x^{(k)}, y^{(k)}, \rho^{(k)})_{k \in K}$  and  $\xi$  as in Lemma 43 and let  $x \Rightarrow y$  denote that configuration  $y$  is reachable from configuration  $x$  on a clique. Then:*

1. *There exists a configuration  $q$  and a sequence  $(z_k)_{k \in K}$  such that for large enough  $k$  we have  $x^{(k)} + q \Rightarrow z_k$  and the counts of states from  $\xi$  are zero in  $z_k$ .*
2. *For any configuration  $q$ , there exist configurations  $q'$  and a sequence  $(z_k)_{k \in K}$  such that for large enough  $k$  we have  $q' + x^{(k)} \Rightarrow q' + z_k + q$  and the counts of states in  $\xi$  are zero in  $z_k$ .*
3. *Let  $\tilde{x}^{(k)}$  be a copy of  $x^{(k)}$  disjoint with  $x^{(k)}$ . For large enough  $k$ , there exists a configuration  $w_k$  such that  $x^{(k)} + \tilde{x}^{(k)} \Rightarrow w_k$  and counts of states from  $\xi$  are zero in  $w_k$ .*

We note that while we use only Lemma 45.3 in the next proof, Lemma 45.3 follows from Lemma 44, Lemma 45.1 and Lemma 45.2; we refer to [9] for the details. With the above, we are now ready to prove the lower bound result.

**Theorem 9** *Let  $\mathcal{P}$  be a constant-state protocol that stabilizes in infinite expected time on any connected graph. Fix a constant  $p > 0$ , and let  $G \sim G_{n,p}^c$ . Then the expected stabilization time of  $\mathcal{P}$  on  $G$  satisfies  $\mathbf{E}[T_{\mathcal{P}}(G)] \in \Omega(n^2)$ .*

**Proof** For the sake of contradiction, suppose that  $\mathcal{P}$  stabilizes in expected  $o(n^2)$  steps in  $G \sim G_{n,p}^c$ . By Lemma 36, as  $G \sim G_{n,p}$ , we have  $T_{\mathcal{P}}(G) \in o(n^2)$  with probability  $1 - o(1)$ . By Lemma 43 and Lemma 45.3, there exists a configuration  $w$

that is reachable on the clique in which all states from  $\xi$  have zero count, where  $\xi$  is some set that intersects every leader generating set.

The only states that may have a non-zero count in  $w$  are sets from  $\Lambda \setminus \xi$ . Since every leader generating set has a non-empty intersection with  $\xi$ , the set  $\Lambda \setminus \xi$  cannot completely contain any leader generating set. Then, by definition of leader generating sets, a leader cannot be generated from  $w$ . Additionally,  $w$  has zero leaders since every leader state is itself a leader generating set. Therefore,  $w$  is a stable reachable configuration without any leaders, a contradiction.  $\square$

## 8 Conclusions

In this work, we performed the first focused investigation of time-space trade-offs in the complexity of leader election on general graphs, in the population model. We provided some of the first time and space-efficient protocols for leader election, and the first time complexity bounds that are tight up to logarithmic factors. We introduced “graphical” variants of classic population protocol techniques, such as information dissemination and approximate phase clocks on the upper bound side, and indistinguishability arguments and surgeries for lower bound results.

Our work leaves open the question of *tight* bounds for both space and time complexity on general graph families, particularly in the case of sparse graphs. Another direction is considering other fundamental problems, such as majority, in the same setting, for which our techniques should prove useful.

## A Proof of Lemma 15

**Proof** For a function  $f : V \rightarrow \mathbb{R}$  let  $f(\bar{x})$  denote the average of  $f$  over all  $\deg(x)$  neighbours  $w \in B(x)$  of  $x$ . Note that for  $u \neq v$ , the expected hitting times satisfies  $\mathbf{H}^{\mathcal{P}}(u, v) = \mathbf{H}^{\mathcal{P}}(\bar{u}, v) + 1$ . Using the fact that the random walks are reversible, it can be shown analogously to Lemma 2 in [47] that the hitting times in the population model satisfy

$$\begin{aligned} \mathbf{H}^{\mathcal{P}}(x, y) + \mathbf{H}^{\mathcal{P}}(y, z) + \mathbf{H}^{\mathcal{P}}(z, x) \\ = \mathbf{H}^{\mathcal{P}}(x, z) + \mathbf{H}^{\mathcal{P}}(z, y) + \mathbf{H}^{\mathcal{P}}(y, x) \end{aligned} \quad (1)$$

for any  $x, y, z \in V$ . To see why, the probability of sampling any sequence of edges  $e_1, \dots, e_k$  that has the walk start from  $x$  and end in  $x$  is  $1/m^k$ . The number of walks  $x \rightsquigarrow y \rightsquigarrow z \rightsquigarrow x$  and  $x \rightsquigarrow z \rightsquigarrow y \rightsquigarrow x$  are equal. Hence, the expected length of both types of walks are equal, which yields (1). The above implies that the relation  $x \leq y$  of the nodes defined by  $x \leq y \Leftrightarrow \mathbf{H}^{\mathcal{P}}(x, y) \leq \mathbf{H}^{\mathcal{P}}(y, x)$  gives a preorder on the

nodes. In particular, there exists a minimum value  $z$  in this preorder such that  $\mathbf{H}^{\mathcal{P}}(x, z) \geq \mathbf{H}^{\mathcal{P}}(z, x)$  for all  $x \in V$ . Fix such a minimum node  $z$ . For any  $x, y \in V$ , define

$$\begin{aligned}\Phi(x, y) &= \mathbf{H}^{\mathcal{P}}(x, y) + \mathbf{H}^{\mathcal{P}}(y, z) - \mathbf{H}^{\mathcal{P}}(z, y) \\ &= \mathbf{H}^{\mathcal{P}}(y, x) + \mathbf{H}^{\mathcal{P}}(x, z) - \mathbf{H}^{\mathcal{P}}(z, x) = \Phi(y, x),\end{aligned}$$

where the second line follows from the identity (1). By choice of  $z$ , the potential  $\Phi(x, y)$  is non-negative. Moreover, the potential satisfies  $\Phi(x, y) = \Phi(\bar{x}, y) + 1 = \Phi(x, \bar{y}) + 1$ .

Now we show that  $\mathbf{M}(x, y) \leq \Phi(x, y)$  for all  $x, y \in V$ , which implies the claim of the lemma, as  $\Phi(x, y) \leq 2 \cdot \mathbf{H}^{\mathcal{P}}(G)$ . For the sake of contradiction, suppose this is not true. Let  $\alpha > 0$  be the maximum value attained by  $\mathbf{M}(x, y) - \Phi(x, y)$  over  $x, y \in V$ . Choose  $x, y$  to be the closest pair of nodes that satisfies  $\mathbf{M}(x, y) = \Phi(x, y) + \alpha$ . Note that  $x \neq y$  and  $\mathbf{M}(x', y) \leq \Phi(x', y) + \alpha$  for all  $x' \in V$ . However, there is some neighbour  $w$  of  $x$  that is closer to  $y$  than  $x$ , so  $\mathbf{M}(w, y) < \Phi(w, y) + \alpha$ . However, this yields a contradiction, as

$$\begin{aligned}\mathbf{M}(x, y) &= 1 + \mathbf{M}(\bar{x}, y) < 1 + \Phi(\bar{x}, y) + \alpha \\ &= \Phi(x, y) + \alpha = \mathbf{M}(x, y).\end{aligned}$$

□

## B Proof of Lemma 19

Let  $h > 0$  be an integer and  $K$  be the number of fair coin flips needed to observe  $h$  consecutive heads. We now show the following technical lemma.

**Lemma 19** *The random variable  $K$  satisfies  $Z_0 \leq K \leq Z_1 + h$ , where  $Z_0 \sim \text{Geom}(2^{-h})$  and  $Z_1 \sim \text{Geom}(2^{-h-1})$ .*

Define  $f(k) = \Pr[K \geq k]$  for all  $k \geq 0$ . Clearly,  $f(k) = 1$  for  $0 \leq k \leq h$ . We will now show that for all  $k \geq h$  the function  $f(k)$  satisfies

$$\left(1 - \frac{1}{2^h}\right)^k \leq f(k) \leq \left(1 - \frac{1}{2^{h+1}}\right)^{k-h}.$$

This then establishes Lemma 19, as the lower bound equals  $\Pr[Z_0 \geq k]$  and the upper bound equals  $\Pr[Z_1 + h \geq k]$ .

**Lemma 46** *For all  $k \geq h$ , we have the identity*

$$f(k+1) = f(k) - \frac{f(k-h)}{2^{h+1}}.$$

**Proof** Recall that  $K$  is the length of the shortest prefix of the infinite sequence  $(X_i)_{i \geq 1}$  of i.i.d. random variables that contains  $h$  consecutive ones, where  $X_i$  denotes if the nodes

was a initiator on its  $i$ th interaction and  $\Pr[X_i = 1] = 1/2$  for all  $i \geq 1$ . Note the function  $f: \mathbb{N} \rightarrow [0, 1]$  is non-increasing. Suppose that  $K = k$ , then the last  $h$  coin flips must have been heads and the  $(h+1)$ th coin flip from end must have been zero. Moreover, the subsequence  $X_1, \dots, X_{k-h-1}$  cannot contain  $h$  consecutive ones. Therefore, for  $k \geq h$ , we have the identity

$$\Pr[K = k] = \frac{1}{2^{h+1}} \cdot \Pr[K \geq k-h] = \frac{f(k-h)}{2^{h+1}}.$$

For all  $k \geq h$ , this gives the recurrence relation

$$f(k+1) = f(k) - \frac{f(k-h)}{2^{h+1}}.$$

□

**Lemma 47** *For all  $k \geq 0$ , the function  $f(k)$  satisfies*

$$\left(1 - \frac{1}{2^h}\right)^k \leq f(k).$$

**Proof** We now show by induction that  $f(k) > f(k-h)/2$  for  $k \geq h$ . As the base case, note that for  $h \leq k \leq 2h$  Lemma 46 gives

$$f(k) = f(h) - \frac{k-h}{2^{h+1}} \geq 1 - \frac{h}{2^{h+1}} > f(k-h)/2,$$

since  $f(i) = 1$  for all  $i \leq h$ . As the induction hypothesis, suppose  $f(k') > f(k'-h)/2$  holds for all  $k \geq k' \geq h$ . The inductive step now follows by observing that for  $k > 2h$  we have

$$\begin{aligned}f(k) &= f(k-1) - \frac{1}{2^{h+1}} f(k-h-1) \\ &= f(k-2) - \frac{1}{2^{h+1}} f(k-h-1) - \frac{1}{2^{h+1}} f(k-h-2) \\ &\quad \dots \\ &= f(k-h) - \frac{1}{2^{h+1}} f(k-h-1) \\ &\quad - \frac{1}{2^{h+1}} f(k-h-2) - \dots - \frac{1}{2^{h+1}} f(k-2h) \\ &\geq f(k-h) - \frac{h}{2^{h+1}} f(k-2h) \\ &> f(k-h) - \frac{h}{2^h} f(k-h) \\ &\geq f(k-h) - \frac{1}{2} f(k-h) = \frac{1}{2} f(k-h),\end{aligned}$$

where the last inequality is given by the induction hypothesis. This completes the induction. Since  $f$  is non-increasing, we

get the that for any  $k > h$

$$f(k+1) = f(k) - \frac{f(k-h)}{2^{h+1}} \geq f(k) \cdot \left(1 - \frac{1}{2^h}\right).$$

Moreover, note that above holds for  $k \leq h$  as well since  $f(k) = 1$  for any  $k \leq h$ . Thus,

$$f(k) \geq \left(1 - \frac{1}{2^h}\right)^k.$$

**Lemma 19** *The random variable  $K$  satisfies  $Z_0 \leq K \leq Z_1 + h$ , where  $Z_0 \sim \text{Geom}(2^{-h})$  and  $Z_1 \sim \text{Geom}(2^{-h-1})$ .*

**Proof** The claim follows using the definition of  $f(k) = \Pr[K \geq k]$  and stochastic domination, since

$$\begin{aligned} \Pr[Z_0 \geq k] &= \left(1 - \frac{1}{2^h}\right)^k \leq f(k) = \Pr[K \geq k] \\ &\leq \left(1 - \frac{1}{2^{h+1}}\right)^{k-h} = \Pr[Z_1 + h \geq k]. \end{aligned}$$

□

□

**Lemma 48** *For all  $0 \leq k \leq h$ , we have  $f(k) = 1$  and for all  $k \geq h$ , the function  $f(k)$  satisfies*

$$f(k) \leq \left(1 - \frac{1}{2^{h+1}}\right)^{k-h}.$$

**Proof** The proof of the upper bound proceeds in a similar fashion. We prove by induction that

$$f(k) \leq \left(1 - \frac{1}{2^{h+1}}\right)^{k-h}$$

for all  $k \geq h$ . For the base case of the induction, note that for  $h \leq k \leq 2h$  we have from Lemma 46 and the fact that  $f(i) = 1$  for  $0 \leq i \leq h$  that

$$f(k) = f(h) - \frac{k-h}{2^{h+1}} = 1 - \frac{k-h}{2^{h+1}} \leq \left(1 - \frac{1}{2^{h+1}}\right)^{k-h},$$

where the inequality follows from the Bernoulli inequality  $1 + rx \leq (1+x)^r$  for  $x \geq -1$  and  $r \geq 0$ . For the inductive step, we get that

$$\begin{aligned} f(k+1) &= f(k+1-h) - \frac{1}{2^{h+1}} \cdot \sum_{i=1}^h f(k+1-h-i) \\ &\leq f(k+1-h) - \frac{h}{2^{h+1}} \cdot f(k+1-h) \\ &\leq f(k+1-h) \cdot \left(1 - \frac{h}{2^{h+1}}\right) \\ &\leq f(k+1-h) \cdot \left(1 - \frac{1}{2^{h+1}}\right)^h \\ &\leq \left(1 - \frac{1}{2^{h+1}}\right)^{k+1-2h} \cdot \left(1 - \frac{1}{2^{h+1}}\right)^h \\ &= \left(1 - \frac{1}{2^{h+1}}\right)^{k+1-h} \end{aligned}$$

by using the Bernoulli inequality and the induction hypothesis. □

**Acknowledgements** We thank all anonymous reviewers for their helpful comments. We would also like to thank Jakob Solnerzik and Olivier Stielert for catching some errors in the proofs.

**Funding** Open Access funding enabled and organized by Projekt DEAL. We gratefully acknowledge funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 805223 ScaleML).

## Declarations

**Competing interests** The authors have no competing interests to declare that are relevant to the content of this article.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Angluin, D.: Local and global properties in networks of processors. In: Proc. of the 12th Annual ACM Symposium on Theory of Computing (STOC), pp. 82–93 (1980)
2. Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M.J., Peralta, R.: Computation in networks of passively mobile finite-state sensors. *Distributed Computing* **18**(4), 235–253 (2006). <https://doi.org/10.1007/s00446-005-0138-3>
3. Aspnes, J., Ruppert, E.: An introduction to population protocols. In: Middleware for Network Eccentric and Mobile Applications, pp. 97–120. Springer, Berlin, Heidelberg (2009). [https://doi.org/10.1007/978-3-540-89707-1\\_5](https://doi.org/10.1007/978-3-540-89707-1_5)
4. Angluin, D., Aspnes, J., Fischer, M.J., Jiang, H.: Self-stabilizing population protocols. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)* **3**(4), 1–28 (2008). <https://doi.org/10.1145/1452001.1452003>

5. Angluin, D., Aspnes, J., Chan, M., Fischer, M.J., Jiang, H., Peralta, R.: Stably computable properties of network graphs. In: International Conference on Distributed Computing in Sensor Systems, pp. 63–74 (2005). [https://doi.org/10.1007/11502593\\_8](https://doi.org/10.1007/11502593_8). Springer
6. Elsässer, R., Radzik, T.: Recent results in population protocols for exact majority and leader election. *Bulletin of the EATCS* **126** (2018)
7. Alistarh, D., Gelashvili, R.: Recent algorithmic advances in population protocols. *SIGACT News* **49**(3), 63–73 (2018). <https://doi.org/10.1145/3289137.3289150>
8. Angluin, D., Aspnes, J., Eisenstat, D.: Fast computation by population protocols with a leader. *Distributed Computing* **21**(3), 183–199 (2008). <https://doi.org/10.1007/s00446-008-0067-z>
9. Doty, D., Soloveichik, D.: Stable leader election in population protocols requires linear time. *Distributed Computing* **31**(4), 257–271 (2018)
10. Alistarh, D., Gelashvili, R.: Polylogarithmic-time leader election in population protocols. In: Proc. 42nd International Colloquium on Automata, Languages, and Programming (ICALP 2015), pp. 479–491 (2015). [https://doi.org/10.1007/978-3-662-47666-6\\_38](https://doi.org/10.1007/978-3-662-47666-6_38)
11. Alistarh, D., Aspnes, J., Eisenstat, D., Gelashvili, R., Rivest, R.L.: Time-space trade-offs in population protocols. In: Proc. 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2017), pp. 2560–2579 (2017). <https://doi.org/10.5555/3039686.3039855>
12. Berenbrink, P., Kaaser, D., Kling, P., Otterbach, L.: Simple and efficient leader election. In: 1st Symposium on Simplicity in Algorithms (SOSA 2018), vol. 61, pp. 9–1911. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2018). <https://doi.org/10.4230/OASIcs.SOSA.2018.9>
13. Gąsieniec, L., Stachowiak, G., Uznański, P.: Almost logarithmic-time space optimal leader election in population protocols. In: Proc. 31st ACM Symposium on Parallelism in Algorithms and Architectures (SPAA 2019) (2019). <https://doi.org/10.1145/3323165.3323178>
14. Gąsieniec, L., Stachowiak, G.: Fast space optimal leader election in population protocols. In: Proc. 29th ACM-SIAM Symposium on Discrete Algorithms (SODA 2018) (2018). <https://doi.org/10.1137/1.9781611975031.169>
15. Gąsieniec, L., Stachowiak, G.: Enhanced phase clocks, population protocols, and fast space optimal leader election. *Journal of the ACM (JACM)* **68**(1), 1–21 (2020). <https://doi.org/10.1145/3424659>
16. Sudo, Y., Ooshita, F., Izumi, T., Kakugawa, H., Masuzawa, T.: Logarithmic expected-time leader election in population protocol model. In: Proc. International Symposium on Stabilizing, Safety, and Security of Distributed Systems (SSS 2019), pp. 323–337 (2019). [https://doi.org/10.1007/978-3-030-34992-9\\_26](https://doi.org/10.1007/978-3-030-34992-9_26)
17. Sudo, Y., Masuzawa, T.: Leader election requires logarithmic time in population protocols. *Parallel Processing Letters* **30**(01), 2050005 (2020). <https://doi.org/10.1142/S012962642050005X>
18. Berenbrink, P., Giakkoupis, G., Kling, P.: Optimal time and space leader election in population protocols. In: Proc. 52nd Annual ACM SIGACT Symposium on Theory of Computing (STOC 2020), pp. 119–129 (2020). <https://doi.org/10.1145/3357713.3384312>
19. Beauquier, J., Blanchard, P., Burman, J.: Self-stabilizing leader election in population protocols over arbitrary communication graphs. In: International Conference on Principles of Distributed Systems (OPODIS), pp. 38–52 (2013). Springer. <https://hal.archives-ouvertes.fr/hal-00867287v2>
20. Sudo, Y., Ooshita, F., Izumi, T., Kakugawa, H., Masuzawa, T.: Time-optimal leader election in population protocols. *IEEE Transactions on Parallel and Distributed Systems* **31**(11), 2620–2632 (2020). <https://doi.org/10.1109/TPDS.2020.2991771>
21. Chen, H.-P., Chen, H.-L.: Self-stabilizing leader election. In: Proc. ACM Symposium on Principles of Distributed Computing (PODC), pp. 53–59 (2019). <https://doi.org/10.1145/3293611.3331616>
22. Chen, H.-P., Chen, H.-L.: Self-stabilizing leader election in regular graphs. In: Proc. 39th Symposium on Principles of Distributed Computing. PODC '20, pp. 210–217. Association for Computing Machinery, New York, NY, USA (2020). <https://doi.org/10.1145/3382734.3405733>
23. Sudo, Y., Masuzawa, T., Datta, A.K., Larmore, L.L.: The same speed timer in population protocols. In: 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS), pp. 252–261 (2016). IEEE
24. Sudo, Y., Ooshita, F., Kakugawa, H., Masuzawa, T.: Loosely stabilizing leader election on arbitrary graphs in population protocols without identifiers or random numbers. *IEICE Transactions on Information and Systems* **103**(3), 489–499 (2020)
25. Sudo, Y., Shibata, M., Nakamura, J., Kim, Y., Masuzawa, T.: Self-stabilizing population protocols with global knowledge. *IEEE Transactions on Parallel and Distributed Systems* **32**(12), 3011–3023 (2021). <https://doi.org/10.1109/TPDS.2021.3076769>
26. Yokota, D., Sudo, Y., Masuzawa, T.: Time-optimal self-stabilizing leader election on rings in population protocols. In: International Symposium on Stabilizing, Safety, and Security of Distributed Systems, pp. 301–316 (2020). Springer
27. Alistarh, D., Gelashvili, R., Rybicki, J.: Fast graphical population protocols. In: Proc. 25th International Conference on Principles of Distributed Systems (OPODIS) (2021). <https://arxiv.org/abs/2102.08808>
28. Peleg, D.: *Distributed Computing: A Locality-Sensitive Approach*. Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, Philadelphia, PA (2000)
29. Kutten, S., Pandurangan, G., Peleg, D., Robinson, P., Trehan, A.: On the complexity of universal leader election. *Journal of the ACM (JACM)* **62**(1), 1–27 (2015). <https://doi.org/10.1145/2699440>
30. Chierichetti, F., Lattanzi, S., Panconesi, A.: Rumour spreading and graph conductance. In: Proc. 21st ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1657–1663 (2010). <https://doi.org/10.1137/1.9781611973075.135>. SIAM
31. Giakkoupis, G., Nazari, Y., Woelfel, P.: How asynchrony affects rumor spreading time. In: Proc. 2016 ACM Symposium on Principles of Distributed Computing (PODC), pp. 185–194 (2016). <https://doi.org/10.1145/2933057.2933117>
32. Levin, D.A., Peres, Y.: *Markov Chains and Mixing Times*, vol. 107. American Mathematical Society, Providence, Rhode Island (2017)
33. Alistarh, D., Rybicki, J., Voitovych, S.: Near-optimal leader election in population protocols on graphs. In: Proceedings of the 2022 ACM Symposium on Principles of Distributed Computing. PODC'22, pp. 246–256. Association for Computing Machinery, New York, NY, USA (2022). <https://doi.org/10.1145/3519270.3538435>
34. Löwe, M., Torres, F.: On hitting times for a simple random walk on dense Erdős-Rényi random graphs. *Statistics & Probability Letters* **89**, 81–88 (2014). <https://doi.org/10.1016/j.spl.2014.02.017>
35. Canonne, C.: A short note on Poisson tail bounds. Accessed January 27, 2022 <http://www.cs.columbia.edu/~ccanonne/files/misc/2017-poissonconcentration.pdf> (2019)
36. Janson, S.: Tail bounds for sums of geometric and exponential variables. *Statistics & Probability Letters* **135**, 1–6 (2018). <https://doi.org/10.1016/j.spl.2017.11.017>
37. Demers, A., Greene, D., Hauser, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D., Terry, D.: Epidemic algorithms for replicated database maintenance. In: Proc. 6th Annual ACM Symposium on Principles of Distributed Computing (PODC), pp. 1–12 (1987). <https://doi.org/10.1145/41840.41841>
38. Acan, H., Collevecchio, A., Mehrabian, A., Wormald, N.: On the push & pull protocol for rumor spreading. *SIAM Journal on Dis-*

crete Mathematics **31**(2), 647–668 (2017). <https://doi.org/10.1137/15M1033113>

- 39. Sauerwald, T.: On mixing and edge expansion properties in randomized broadcasting. *Algorithmica* **56**(1), 51–88 (2010). <https://doi.org/10.1007/s00453-008-9245-4>
- 40. Karp, R., Schindelhauer, C., Shenker, S., Vocking, B.: Randomized rumor spreading. In: Proc. 41st Annual Symposium on Foundations of Computer Science (FOCS 2000), pp. 565–574 (2000)
- 41. Chierichetti, F., Lattanzi, S., Panconesi, A.: Almost tight bounds for rumour spreading with conductance. In: Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC), pp. 399–408 (2010). <https://doi.org/10.1145/1806689.1806745>
- 42. Fatès, N., Gerin, L.: Examples of fast and slow convergence of 2d asynchronous cellular systems. In: International Conference on Cellular Automata, pp. 184–191 (2008). [https://doi.org/10.1007/978-3-540-79992-4\\_24](https://doi.org/10.1007/978-3-540-79992-4_24). Springer
- 43. Ottino-Löffler, B., Scott, J.G., Strogatz, S.H.: Takeover times for a simple model of network infection. *Physical Review E* **96**(1), 012313 (2017). <https://doi.org/10.1103/PhysRevE.96.012313>
- 44. Liggett, T.M.: Interacting Particle Systems. Springer, Berlin, Heidelberg (2005). <https://doi.org/10.1007/b138374>
- 45. Giakkoupis, G.: Tight bounds for rumor spreading in graphs of a given conductance. In: Schwentick, T., Dürr, C. (eds.) 28th International Symposium on Theoretical Aspects of Computer Science (STACS 2011). Leibniz International Proceedings in Informatics (LIPIcs), vol. 9, pp. 57–68. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2011). <https://doi.org/10.4230/LIPIcs.STACS.2011.57>. <http://drops.dagstuhl.de/opus/volltexte/2011/2997>
- 46. Hoffman, C., Kahle, M., Paquette, E.: Spectral gaps of random graphs and applications. *International Mathematics Research Notices* (2019)
- 47. Coppersmith, D., Tetali, P., Winkler, P.: Collisions among random walks on a graph. *SIAM Journal on Discrete Mathematics* **6**(3), 363–374 (1993). <https://doi.org/10.1137/0406029>
- 48. Vershynin, R.: High-dimensional Probability: An Introduction with Applications in Data Science, vol. 47. Cambridge University Press, Cambridge, United Kingdom (2018)
- 49. Bollobás, B.: Random Graphs, 2nd edn. Cambridge University Press, Cambridge, United Kingdom (2001)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.