

Symmetry Breaking in the Congest Model: Time- and Message-Efficient Algorithms for Ruling Sets*

Shreyas Pai¹, Gopal Pandurangan², Sriram V. Pemmaraju¹, Talal Riaz¹, and Peter Robinson³

- 1 Department of Computer Science, The University of Iowa, Iowa City, IA 52242, USA. E-mail: {shreyas-pai, sriram-pemmaraju, talal-riaz}@uiowa.edu. Supported in part by NSF grant CCF-1318166.
- 2 Department of Computer Science, University of Houston, Houston, TX 77204, USA. E-mail: gopalpandurangan@gmail.com. Supported, in part, by NSF grants CCF-1527867, CCF-1540512, IIS-1633720, and CCF-1717075.
- 3 Department of Computer Science, Royal Holloway, University of London, UK. E-mail: peter.robinson@rhul.ac.uk.

Abstract

We study local symmetry breaking problems in the CONGEST model, focusing on ruling set problems, which generalize the fundamental Maximal Independent Set (MIS) problem. The *time (round) complexity* of MIS (and ruling sets) have attracted much attention in the LOCAL model. Indeed, recent results (Barenboim et al., FOCS 2012, Ghaffari SODA 2016) for the MIS problem have tried to break the long-standing $O(\log n)$ -round “barrier” achieved by Luby’s algorithm, but these yield $o(\log n)$ -round complexity only when the maximum degree Δ is somewhat small relative to n . *More importantly, these results apply only in the LOCAL model.* In fact, the best known time bound in the CONGEST model is still $O(\log n)$ (via Luby’s algorithm) even for moderately small Δ (i.e., for $\Delta = \Omega(\log n)$ and $\Delta = o(n)$). Furthermore, *message complexity* has been largely ignored in the context of local symmetry breaking. Luby’s algorithm takes $O(m)$ messages on m -edge graphs and this is the best known bound with respect to messages. Our work is motivated by the following central question: can we break the $\Theta(\log n)$ time complexity barrier and the $\Theta(m)$ message complexity barrier in the CONGEST model for MIS or closely-related symmetry breaking problems?

This paper presents progress towards this question for the distributed ruling set problem in the CONGEST model. A β -*ruling set* is an independent set such that every node in the graph is at most β hops from a node in the independent set. We present the following results:

- *Time Complexity:* We show that we can break the $O(\log n)$ “barrier” for 2- and 3-ruling sets. We compute 3-ruling sets in $O\left(\frac{\log n}{\log \log n}\right)$ rounds with high probability (whp). More generally we show that 2-ruling sets can be computed in $O\left(\log \Delta \cdot (\log n)^{1/2+\varepsilon} + \frac{\log n}{\log \log n}\right)$ rounds for any $\varepsilon > 0$, which is $o(\log n)$ for a wide range of Δ values (e.g., $\Delta = 2^{(\log n)^{1/2-\varepsilon}}$). These are the first 2- and 3-ruling set algorithms to improve over the $O(\log n)$ -round complexity of Luby’s algorithm in the CONGEST model.
- *Message Complexity:* We show an $\Omega(n^2)$ lower bound on the message complexity of computing an MIS (i.e., 1-ruling set) which holds also for randomized algorithms and present a contrast to this by showing a randomized algorithm for 2-ruling sets that, whp, uses only $O(n \log^2 n)$ messages and runs in $O(\Delta \log n)$ rounds. This is the first message-efficient algorithm known for ruling sets, which has message complexity nearly linear in n (which is optimal up to a polylogarithmic factor).

* A short version of this paper appeared as a 3-page “Brief Announcement” in PODC 2017. A full version of this paper appears on arxiv [18].



1998 ACM Subject Classification C.2.4 Distributed Systems, F.1.2 Modes of Computation, F.2.2 Nonnumerical Algorithms and Problems, G.2.2 Graph Theory

Keywords and phrases Congest model; Local model; Maximal independent set, Message complexity; Round complexity; Ruling sets; Symmetry breaking

Digital Object Identifier 10.4230/LIPIcs.DISC.2017.38

1 Introduction

The *maximal independent set (MIS)* problem is one of the fundamental problems in distributed computing because it is an elegant abstraction of “local symmetry breaking,” an issue that arises repeatedly in distributed computing. About 30 years ago Alon, Babai, and Itai [1] and Luby [17] presented a randomized algorithm for MIS, running on n -node graphs in $O(\log n)$ rounds with high probability (whp)¹. Since then the MIS problem has been studied extensively and recently, there has been some exciting progress in designing faster MIS algorithms. For n -node graphs with maximum degree Δ , Ghaffari [10] presented an MIS algorithm running in $O(\log \Delta) + 2^{O(\sqrt{\log \log n})}$ rounds, improving over the algorithm of Barenboim et al. [5] that runs in $O(\log^2 \Delta) + 2^{O(\sqrt{\log \log n})}$ rounds. Ghaffari’s MIS algorithm is the first MIS algorithm to improve over the round complexity of Luby’s algorithm when $\Delta = 2^{o(\log n)}$ and Δ is bounded below by $\Omega(\log n)$.²

While the results of Ghaffari and Barenboim et al. constitute a significant improvement in our understanding of the round complexity of the MIS problem, it should be noted that both of these results are in the LOCAL model. The LOCAL model [20] is a synchronous, message-passing model of distributed computing in which *messages can be arbitrarily large*. Luby’s algorithm, on the other hand, is in the CONGEST model [20] and uses small messages, i.e., messages that are $O(\log n)$ bits or $O(1)$ words in size. In fact, to date, *Luby’s algorithm is the fastest known MIS algorithm in the CONGEST model*; this is the case even when Δ is between $\Omega(\log n)$ and $2^{o(\log n)}$. For example, for the class of graphs with $\Delta = 2^{O(\sqrt{\log n})}$, Ghaffari’s MIS algorithm runs in $O(\sqrt{\log n})$ rounds whp in the LOCAL model, but we don’t know how to compute an MIS for this class of graphs in $o(\log n)$ rounds in the CONGEST model. It should be further noted that the MIS algorithms of Ghaffari and Barenboim et al. use messages of size $O(\text{poly}(\Delta) \log n)$ (see Theorem 3.5 in [5]), which can be much larger than the $O(\log n)$ -sized messages allowed in the CONGEST model; in fact these algorithms do not work even if messages of size $O(\text{poly}(\log n))$ were allowed. Furthermore, large messages arise in these algorithms from a topology-gathering step in which cluster-leaders gather the entire topology of their clusters in order to compute an MIS of their cluster – this step seems fundamental to these algorithms and there does not seem to be an efficient way to simulate this step in the CONGEST model.

Ruling sets are a natural generalization of MIS and have also been well-studied in the LOCAL model. An (α, β) -*ruling set* [11] is a node-subset T such that (i) any two distinct nodes in T are at least α hops apart in G and (ii) every node in the graph is at most β hops from some node in T . A $(2, \beta)$ -ruling set is an independent set and since such ruling sets

¹ Throughout, we use “with high probability (whp)” to mean with probability at least $1 - 1/n^c$, for some $c \geq 1$.

² For $\Delta = o(\log n)$, the deterministic MIS algorithm of Barenboim, Elkin, and Kuhn [3] that runs $O(\Delta + \log^* n)$ rounds is faster than Luby’s algorithm.

are the focus of this paper, we use the shorthand β -ruling sets to refer to $(2, \beta)$ -ruling sets. (Using this terminology an MIS is just a 1-ruling set.) The above mentioned MIS results [5, 10] have also led to the *sublogarithmic*-round algorithms for β -ruling sets for $\beta \geq 2$. The earliest instance of such a result was the algorithm of Kothapalli and Pemmaraju [14] that computed a 2-ruling set in $O(\sqrt{\log \Delta} \cdot (\log n)^{1/4})$ rounds by using an earlier version of the Barenboim et al. [4] MIS algorithm. There have been several further improvements in the running time of ruling set algorithms culminating in the $O(\beta \log^{1/\beta} \Delta) + 2^{O(\sqrt{\log \log n})}$ round β -ruling set algorithm of Ghaffari [10]. This result is based on a recursive sparsification procedure of Bisht et al. [7] that reduces the β -ruling set problem on graphs with maximum degree Δ to an MIS problem on graphs with degree much smaller. Ghaffari's β -ruling set result is also interesting because it identifies a separation between 2-ruling sets and MIS (1-ruling sets). This follows from the lower bound of $\Omega\left(\min\left\{\sqrt{\frac{\log n}{\log \log n}}, \frac{\log \Delta}{\log \log \Delta}\right\}\right)$ for MIS due to Kuhn et al. [15]. Again, we emphasize here that all of these improvements for ruling set algorithms are *only in the LOCAL model* because these ruling set algorithms rely on LOCAL-model MIS algorithms to “finish off” the processing of small degree subgraphs. As far as we know, prior to the current work there has been no $o(\log n)$ -round, β -ruling set algorithm in the CONGEST model for any $\beta = O(1)$.

The focus of all the above results has been on the time (round) complexity. *Message complexity*, on the other hand, has been largely ignored in the context of local symmetry breaking problems such as MIS and ruling sets. For a graph with m edges, Luby's algorithm uses $O(m)$ messages in the CONGEST model and until now there has been no MIS or ruling set algorithm that uses $o(m)$ messages. We note that the ruling set algorithm of Goldberg et al. [11] which can be implemented in the CONGEST model [12] also takes at least $\Omega(m)$ messages.

The focus of this paper is symmetry breaking problems in the CONGEST model and the specific question that motivates our work is whether we can go beyond Luby's algorithm in the CONGEST model for MIS or any closely-related symmetry breaking problems such as *ruling sets*. In particular, *can we break the $\Theta(\log n)$ time complexity barrier and the $\Theta(m)$ message complexity barrier, in the CONGEST model for MIS and ruling sets?* In many applications, especially in resource-constrained communication networks and in distributed processing of large-scale data [13], it is important to design distributed algorithms that have low time complexity as well as message complexity.

We present two sets of results, one set focusing on time (round) complexity and the other on message complexity.

1. Time complexity: (cf. Section 2) We first show that 2-ruling sets can be computed in the CONGEST model in $O\left(\log \Delta \cdot (\log n)^{1/2+\varepsilon} + \frac{\log n}{\log \log n}\right)$ rounds whp for n -node graphs with maximum degree Δ and for any $\varepsilon > 0$. This is the first algorithm to improve over Luby's algorithm, by running in $o(\log n)$ rounds in the CONGEST model, for a wide range of values of Δ . Specifically our algorithm runs in $o(\log n)$ rounds for Δ bounded above by $2^{(\log n)^{1/2-\varepsilon}}$ for any value of $\varepsilon > 0$. In the full version [18], we show how to compute 3-ruling sets (for any graph) in $O\left(\frac{\log n}{\log \log n}\right)$ rounds whp in the CONGEST model, using our 2-ruling set algorithm as a subroutine. We also present a simple 5-ruling set algorithm based on Ghaffari's MIS algorithm that runs in $O(\sqrt{\log n})$ rounds in the CONGEST model.

2. Message complexity: (cf. Sections 3 and 4) We show that $\Omega(n^2)$ is a fundamental lower bound for computing an MIS (i.e., 1-ruling set) by showing that there exists graphs (with $m = \Theta(n^2)$ edges) where any distributed MIS algorithm needs $\Omega(n^2)$ messages. In contrast, we show that 2-ruling sets can be computed using significantly smaller message complexity.

In particular, we present a randomized 2-ruling set algorithm that, whp, uses $O(n \log^2 n)$ messages and runs in $O(\Delta \log n)$ rounds. This is the first $o(m)$ -message algorithm known for ruling sets, which takes near-linear (in n) message complexity. This message bound is tight up to a polylogarithmic factor, since we show that any $O(1)$ -ruling set (randomized) algorithm that succeeds with probability $1 - o(1)$ requires $\Omega(n)$ messages in the worst case. We also present a simple 2-ruling set algorithm that uses $O(n^{1.5} \log n)$ messages, but runs faster — in $O(\log n)$ rounds.

Our results make progress towards understanding the complexity of symmetry breaking, in particular with respect to ruling sets, in the CONGEST model. With regards to time complexity, our results, for the first time, show that one can obtain $o(\log n)$ round algorithms for ruling sets in the CONGEST model. With regards to message complexity, our results are (essentially) tight: while MIS needs quadratic (in n) messages in the worst case, 2-ruling sets can be computed using near-linear (in n) messages. Other related work and omitted proofs can be found in the full version [18].

1.1 Distributed Computing Model

We consider the standard synchronous CONGEST model [20] described as follows.

We are given a distributed network of n nodes, modeled as an undirected graph G . Each node hosts a processor with limited initial knowledge. We assume that nodes have unique IDs (this is not essential, but simplifies presentation), and at the beginning of the computation each node is provided its ID as input. Thus, a node has only *local* knowledge³. Specifically we assume that each node has ports (each port having a unique port number); each incident edge is connected to one distinct port. This model is referred to as the *clean network model* in [20] and is also sometimes referred to as the KT_0 model, i.e., the initial (K)nowledge of all nodes is restricted (T)ill radius 0 (i.e., just the local knowledge) [2].

Nodes are allowed to communicate through the edges of the graph G and it is assumed that communication is synchronous and occurs in discrete rounds (time steps). In each round, each node can perform some local computation including accessing a private source of randomness, and can exchange (possibly distinct) $O(\log n)$ -bit messages with each of its neighboring nodes. This model of distributed computation is called the CONGEST($\log n$) model or simply the CONGEST model [20].

1.2 Technical Overview

1.2.1 Time Bounds

The MIS algorithms of Barenboim et al. [5] and Ghaffari [10] use a 2-phase strategy, attributed to Beck [6], who used it in his algorithmic version of the Lovász Local Lemma. In the first phase, some number of iterations of a Luby-type “base algorithm” are run (in the CONGEST model). During this phase, some nodes join the MIS and these nodes and their neighbors become inactive. The first phase is run until the graph is “shattered”, i.e., the nodes that remain active induce a number of “small” connected components. Once the graph is “shattered”, the algorithm switches to the second, deterministic phase to “finish off” the

³ Our near-linear message-efficient algorithm (Section 3) does not require knowledge of n or Δ , whereas our time-efficient algorithms (Section 2) assume knowledge of n and Δ (otherwise it will work up to a given Δ).

problem in the remaining small components. It is this second phase that relies critically on the use of the LOCAL model in order to run fast.

In general, in the CONGEST model it is not clear how to take advantage of low degree or low diameter or small size of a connected component to solve symmetry-breaking problems (MIS or ruling sets) faster than the $O(\log n)$ -round bound provided by Luby's algorithm. In both Barenboim et al. [5] and Ghaffari [10], a key ingredient of the second "finish-off" phase is the deterministic network decomposition algorithm of Panconesi and Srinivasan [19] that can be used to compute an MIS in $O(2\sqrt{\log s})$ rounds on a graph with s nodes in the LOCAL model. If one can get connected components of size $O(\text{poly}(\log n))$ then it is possible to finish the rest of the algorithm in $2^{O(\sqrt{\log \log n})}$ rounds and this is indeed the source of the " $2^{O(\sqrt{\log \log n})}$ " term in the round complexity of these MIS algorithms. In fact, the Panconesi-Srinivasan network decomposition algorithm itself runs in the CONGEST model, but once the network has been decomposed into small diameter clusters then algorithms simply resort to gathering the entire topology of a cluster at a cluster-leader and this requires large messages. Currently, there seem to be no techniques for symmetry breaking problems in the CONGEST model that are able to take advantage of the diameter of a network being small. As far as we know, there is no $o(\log n)$ -round $O(1)$ -ruling set algorithm in the CONGEST model *even for constant-diameter graphs*, for any constant larger than 1. To obtain our sublogarithmic β -ruling set algorithms (for $\beta = 2, 3, 5$), we use simple *greedy* MIS and 2-ruling set algorithms to process "small" subgraphs in the final stages of algorithm. These greedy algorithms just exchange $O(\log n)$ -bit IDs with neighbors and run in the CONGEST model, but they can take $\Theta(s)$ rounds in the worst case, where s is the length of the *longest* path in the subgraph. So our main technical contribution is to show that it is possible to do a randomized shattering of the graph so that none of the fragments have any long paths.

1.2.2 Message Bounds

As mentioned earlier, our message complexity lower bound for MIS and the contrasting upper bound for 2-ruling set show a clear separation between these two problems. At a high-level, our lower bound argument exploits the idea of "bridge crossing" (similar to [16]) whose intuition is as follows. We consider two types of related graphs: (1) a *complete bipartite graph* and (2) a *random bridge graph* which consists of two (almost-)complete bipartite graphs connected by two "bridge" edges chosen randomly (see Figure 1 and Section 4 for a detailed description of the construction). Note that the MIS in a complete bipartite graph is exactly the set of all nodes belonging to one part of the partition. The crucial observation is that if no messages are sent over bridge edges, then the bipartite graphs on either side of the bridge edges behave identically which can result in choosing adjacent nodes in MIS, a violation. In particular, we show that if an algorithm sends $o(n^2)$ messages, then with probability at least $1 - o(1)$ that there will be *no* message sent over the bridge edges and by symmetry, with probability at least $1/2$, two nodes that are connected by the bridge edge will be chosen to be in the MIS.

Our 2-ruling set algorithm with low-message-complexity crucially uses the fact that, unlike in an MIS, in a 2-ruling set there are 3 categories of nodes: CATEGORY-1 (nodes that are in the independent set), CATEGORY-2 (nodes that are neighbors of CATEGORY-1) and CATEGORY-3 nodes (nodes that are neighbors of CATEGORY-2, but not neighbors of CATEGORY-1). Our algorithm, inspired by Luby's MIS algorithm, uses three main ideas. First, CATEGORY-2 and CATEGORY-3 nodes don't initiate messages; only undecided nodes (i.e., nodes whose category are not yet decided) initiate messages. Second, an undecided

node does “checking sampling” (cf. Algorithm 3) first before it does local broadcast, i.e., it samples a few of its neighbors to see if there are any CATEGORY-2 nodes; if so it becomes a CATEGORY-3 node immediately. Third, an undecided node tries to enter the ruling set with probability that is *always* inversely proportional to its *original* degree, i.e., $\Theta(1/d(v))$, where $d(v)$ is the degree of v . This is unlike in Luby’s algorithm, where the marking probability is inversely proportional to its *current degree*. These ideas along with an *amortized* charging argument [8] yield our result: an algorithm using $O(n \log^2 n)$ messages and running in $O(\Delta \log n)$ rounds.

2 Time-Efficient Ruling Set Algorithms in the Congest model

The main result of this section is a 2-ruling set algorithm in the CONGEST model that runs in $O\left(\log \Delta \cdot (\log n)^{1/2+\varepsilon} + \frac{\log n}{\log \log n}\right)$ rounds whp, for any constant $\varepsilon > 0$, on n -node graphs with maximum degree Δ . An implication of this result is that for graphs with $\Delta = 2^{O((\log n)^{1/2-\varepsilon})}$ for any $\varepsilon > 0$, we can compute a 2-ruling set in $O\left(\frac{\log n}{\log \log n}\right)$ rounds in the CONGEST model. A second implication is that using this 2-ruling set algorithm as a subroutine, we can compute a 3-ruling set for *any* graph in $O\left(\frac{\log n}{\log \log n}\right)$ rounds whp in the CONGEST model. These are the first sublogarithmic-round CONGEST model algorithms for 2-ruling sets (for a wide range of Δ) and 3-ruling sets. Combining some of the techniques used in these algorithms with the first phase of Ghaffari’s MIS algorithm [10], we show that a 5-ruling set can be computed in $O(\sqrt{\log n})$ rounds whp in the CONGEST model. The 3-ruling set and 5-ruling set results appear in the full paper [18].

2.1 The 2-ruling Set Algorithm

Our 2-ruling set algorithm (described in pseudocode below) takes as input an n -node graph with maximum degree $\Delta \leq 2^{\sqrt{\log n}}$, along with a parameter $\varepsilon > 0$. For $\Delta > 2^{\sqrt{\log n}}$, we simply execute Luby’s MIS algorithm to solve the problem. The algorithm consists of $\lceil \log \Delta \rceil$ *scales* and in scale t , $1 \leq t \leq \lceil \log \Delta \rceil$, nodes with degrees at most $\Delta_t := \Delta/2^{t-1}$ are processed. Each scale consists of $\Theta(\log^{1/2+\varepsilon} n)$ *iterations*. In an iteration i , in scale t , each undecided node independently joins a set $M_{i,t}$ with probability $1/(\Delta_t \cdot \log^\varepsilon n)$ (Line 5). Neighbors of nodes in $M_{i,t}$, that are themselves not in $M_{i,t}$, are set aside and placed in a set $W_{i,t}$ (Lines 6-8). The nodes in $M_{i,t} \cup W_{i,t}$ have decided their fate and we continue to process the undecided nodes. At the end of all the iterations in a scale t , any undecided node that still has $\Delta_t/2$ or more undecided neighbors is placed in a “bad” set B_t for that scale (Line 11), thus effectively deciding the fate of all nodes with degree at least $\Delta_t/2$. We now process the set of scale- t “bad” nodes, B_t , by simply running a greedy 2-ruling set algorithm on B_t (Line 13). We also need to process the sets $M_{i,t}$ (Line 15) and for that we rely on a greedy 1-ruling set algorithm (i.e., a greedy MIS algorithm). Note that the $M_{i,t}$ ’s are all disconnected from each other since the $W_{i,t}$ ’s act as “buffers” around the $M_{i,t}$ ’s. Thus after all the scales are completed, we can compute an MIS on all of the $M_{i,t}$ ’s in parallel. Since each node in $W_{i,t}$ has a neighbor in $M_{i,t}$, this will guarantee that every node in $W_{i,t}$ has an independent set node at most 2 hops away. In the following algorithm we use $\deg_S(v)$ to denote the degree of a vertex v in the $G[S]$, the graph induced by S .

Algorithm 1: 2-RULING SET(Graph $G = (V, E)$, $\varepsilon > 0$):

```

1  $I \leftarrow \emptyset$ ;  $S \leftarrow V$ ;
2 for scale  $t = 1, 2, \dots, \lceil \log \Delta \rceil$  do
3   Let  $\Delta_t = \frac{\Delta}{2^{t-1}}$ ;  $S_t \leftarrow S$ ;
4   for iteration  $i = 1, 2, \dots, \lceil c \cdot \log^{1/2+\varepsilon} n \rceil$  do
5     Each  $v \in S$  marks itself and joins  $M_{i,t}$  with probability  $\frac{1}{\Delta_t \cdot \log^\varepsilon n}$ ;
6     if  $v \in S$  is unmarked and a neighbor in  $S$  is marked then
7       |  $v$  joins  $W_{i,t}$ ;
8     end
9      $S \leftarrow S \setminus (M_{i,t} \cup W_{i,t})$ ;
10  end
11   $B_t \leftarrow \{v \in S \mid \text{deg}_S(v) \geq \Delta_t/2\}$ ;
12   $S \leftarrow S \setminus B_t$ ;
13   $I \leftarrow I \cup \text{GREEDYRULINGSET}(G[S_t], B_t, 2)$ ;
14 end
15  $I \leftarrow I \cup (\cup_t \cup_i \text{GREEDYRULINGSET}(G[S_t], M_{i,t}, 1))$ ;
16 return  $I$ ;

```

The overall round complexity of this algorithm critically depends on the greedy 2-ruling set algorithm terminating quickly on each B_t (Line 13) and the greedy 1-ruling set algorithm terminating quickly on each $M_{i,t}$ (Line 15). To be concrete, we present below a specific β -ruling set algorithm that greedily picks nodes by their IDs from a given node subset R .

Algorithm 2: GREEDYRULINGSET(Graph $G = (V, E)$, $R \subseteq V$, integer $\beta > 0$):

```

1  $I \leftarrow \emptyset$ ;  $U \leftarrow R$ ; //  $U$  is the initial set of undecided nodes
2 while  $U \neq \emptyset$  do
3   for each node  $v \in U$  in parallel do
4     | if ( $v$  has higher ID than all neighbors in  $U$ ) then
5       | |  $I \leftarrow I \cup \{v\}$ ;
6       | |  $v$  and nodes within distance  $\beta$  in  $G$  are removed from  $U$ 
7     | end
8   end
9 end
10 return  $I$ 

```

To show that the calls to this greedy ruling set algorithm terminate quickly, we introduce the notion of *witness paths*. If $\text{GREEDYRULINGSET}(G, R, \beta)$ runs for p iterations (of the **while**-loop), then R must contain a sequence of nodes (v_1, v_2, \dots, v_p) such that v_i , $1 \leq i \leq p$, joins the independent set I in iteration i and node v_i , $1 < i \leq p$, must contain an undecided node with higher ID in its 1-neighborhood in G , which was removed when v_{i-1} and its β -neighborhood in G were removed in iteration $i - 1$. We call such a sequence a *witness path* for the execution of GREEDYRULINGSET . Three simple properties of witness paths are needed in our analysis: (i) any two nodes v_i and v_j in the witness path are at least $\beta + 1$ hops away in G , (ii) any two consecutive nodes v_i and v_{i+1} in the witness path are at most $\beta + 1$ hops away in G , and (iii) $G[R]$ contains a simple path with $(p - 1)(\beta + 1) + 1$ nodes, starting at node v_1 , passing through nodes v_2, v_3, \dots, v_{p-1} and ending at node v_p .

To show that each $M_{i,t}$ can be processed quickly by the greedy 1-ruling set algorithm we show (in Lemma 1) that whp every witness path for the execution of the greedy 1-ruling set algorithm is short. Similarly, to show that each B_t can be processed quickly by the greedy 2-ruling set algorithm we prove (in Lemma 2) that whp a “bad” set B_t cannot contain a witness path of length $\sqrt{\log n}$ or longer to the execution of the greedy 2-ruling set algorithm. At the start of our analysis we observe that the set S_t , which is the set of undecided nodes at the start of scale t , induces a subgraph with maximum degree $\Delta_t = \Delta/2^{t-1}$.

► **Lemma 1.** *For all scales t and iterations i , $\text{GREEDYRULINGSET}(G[S_t], M_{i,t}, 1)$ runs in $O\left(\frac{\log n}{\varepsilon \log \log n}\right)$ rounds, whp.*

Proof. Consider an arbitrary scale t and iteration i . By Property (iii) of witness paths, there is a simple path P with $(2p - 1)$ nodes in $G[S_t]$, all of whose nodes have joined $M_{i,t}$. Due to independence of the marking step (Line 5) the probability that all nodes in P join $M_{i,t}$ is at most $(1/\Delta_t \cdot \log^\varepsilon n)^{2p-1}$. Since $\Delta(G[S_t]) \leq \Delta_t$, the number of simple paths with $2p - 1$ nodes in $G[S_t]$ is at most $n \cdot \Delta_t^{2p-1}$. Using a union bound over all candidate simple paths with $2p - 1$ nodes in $G[S_t]$, we see that the probability that there exists a simple path in $G[M_{i,t}]$ of length $2p - 1$ is at most: $n \cdot \Delta_t^{2p-1} \cdot \left(\frac{1}{\Delta_t \log^\varepsilon n}\right)^{2p-1} = n \cdot \frac{1}{(\log n)^{\varepsilon(2p-1)}}$. Picking p to be the smallest integer such that $2p - 1 \geq \frac{4 \log n}{\varepsilon \log \log n}$, we get

$$\Pr(\exists \text{ a simple path with } 2p - 1 \text{ nodes that joins } M_{i,t}) \leq n \cdot \frac{1}{(2 \log \log n)^{\varepsilon \frac{4 \log n}{\varepsilon \log \log n}}} = n \cdot \frac{1}{n^4} = \frac{1}{n^3}.$$

We have $O(\log \Delta \cdot (\log n)^{1/2+\varepsilon})$ different $M_{i,t}$'s. Using a union bound over these $M_{i,t}$'s, we see that the probability that there exists an $M_{i,t}$ containing a simple path with $2p - 1$ nodes is at most n^{-2} . Thus with probability at least $1 - 1/n^2$, all of the calls to $\text{GREEDYRULINGSET}(G[S_t], M_{i,t}, 1)$ (in Line 15) complete in $O\left(\frac{\log n}{\log \log n}\right)$ rounds. ◀

► **Lemma 2.** *For all scales t , the call to $\text{GREEDYRULINGSET}(G[S_t], B_t, 2)$ takes $O(\sqrt{\log n})$ rounds whp.*

Proof. Let P be a length- p witness path for the execution of $\text{GREEDYRULINGSET}(G[S_t], B_t, 2)$ (Line 13). By Property (i) of witness paths, all pairs of nodes in P are at distance at least 3 from each other. Fix a scale t . We now calculate the probability that all nodes in P belong to B_t . Consider some node $v \in P$. For v to belong to B_t , it must have not marked itself in all iterations of scale t and moreover at least $\Delta_t/2$ neighbors of v in S_t must not have marked themselves in any iteration of scale t . Since the neighborhoods of any two nodes in P are disjoint, the event that v joins B_t is independent of any other node in P joining B_t . Therefore,

$$\Pr(P \text{ is in } B_t) \leq \prod_{v \in P} \Pr(v \text{ and at least } \Delta_t/2 \text{ neighbors do not mark themselves in scale } t).$$

This can be bounded above by $\prod_{v \in P} \left(1 - \frac{1}{\Delta_t (\log n)^\varepsilon}\right)^{\frac{\Delta_t}{2} \cdot c (\log n)^{1/2+\varepsilon}} \leq \exp\left(-\frac{c}{2} \cdot (\log n)^{1/2} \cdot p\right)$.

Plugging in $p = \sqrt{\log n}$ we see that this probability is bounded above by $n^{-c/2}$. By Property (ii) of witness paths and the fact that $\Delta(G[S_t]) \leq \Delta_t$, we know that there are at most $n \cdot (\Delta_t)^{3p}$ length- p candidate witness paths. Using a union bound over all of these, we get that the probability that there exists a witness path that joins B_t is at most $n \Delta_t^{3p} \cdot n^{-c/2}$. Plugging in $\Delta \leq 2\sqrt{\log n}$ and $p = \sqrt{\log n}$ we get that this probability is at most $n \cdot n^3 \cdot n^{-c/2} = n^{-c/2+4}$. Picking a large enough constant c guarantees that this probability is at most $1/n^2$ and taking a final union bound over all $\lceil \log \Delta \rceil$ scales gives us the result that all calls to $\text{GREEDYRULINGSET}(G[S_t], B_t, 2)$ take $O(\sqrt{\log n})$ rounds whp. ◀

Lemmas 1 and 2 prove upper bounds on the number of rounds it takes for the calls to GREEDYRULINGSET (in Lines 13 and 15). Now analyzing Algorithm 2-RULING SET is straightforward and leads to the following theorem.

► **Theorem 3.** *Algorithm 2-RULING SET computes a 2-ruling set in the CONGEST model in $O\left(\log \Delta \cdot (\log n)^{1/2+\varepsilon} + \frac{\log n}{\varepsilon \log \log n}\right)$ rounds, whp.*

3 A Message-Efficient Algorithm for 2-Ruling Set

In this section, we present a randomized distributed algorithm for computing a 2-ruling set in the CONGEST model that takes $O(n \log^2 n)$ messages and $O(\Delta \log n)$ rounds whp, where n is the number of nodes and Δ is the maximum node degree. The algorithm does not require any global knowledge, including knowledge of n or Δ . We show in Theorem 9 that the algorithm is essentially message-optimal (up to a polylog(n) factor). This is the first message-efficient algorithm known for 2-ruling set, i.e., it takes $o(m)$ messages, where m is the number of edges in the graph.⁴

3.1 The Algorithm

Algorithm 3 is inspired by Luby’s algorithm for MIS [17]; however, there are crucial differences. (Note that Luby’s algorithm sends $\Theta(m)$ messages.) Given a ruling set R , we classify nodes in V into three categories:

- CATEGORY-1: nodes that belong to the ruling set R ;
- CATEGORY-2: nodes that have a neighbor in R ; and
- CATEGORY-3: the rest of the nodes, i.e., nodes that have a neighbor in CATEGORY-2.

At the beginning of the algorithm, each node is UNDECIDED, i.e., its category is not set and upon termination, each node knows its category.

Let us describe one iteration of the algorithm (Steps 3-19) from the perspective of an arbitrary node v . Each undecided node v marks itself with probably $1/2d(v)$. If v is marked it samples a set of $\Theta(\log(d(v)))$ random neighbours and checks whether any of them belong to CATEGORY-2 — we call this the *checking sampling step*. If so, then v becomes a CATEGORY-3 node and is done (i.e., it will never broadcast again, but will continue to answer checking sampling queries, if any, from its neighbors). Otherwise, v performs the *broadcast step*, i.e., it communicates with all its neighbors and checks if there is a marked neighbor that is of equal or higher degree, and if so, it unmarks itself; else it enters the ruling set and becomes a CATEGORY-1 node.⁵ Then node v informs all its neighbors about its CATEGORY-1 status causing them to become CATEGORY-2 nodes (if they are not already) and they are done.

A node that does not hear from any of its neighbors knows that it is not a neighbor of any CATEGORY-1 node. Note that CATEGORY-2 and CATEGORY-3 nodes do not initiate messages, which is important for keeping the message complexity low. Another main idea in reducing messages is the random sampling check of a few neighbors to see whether any of them are CATEGORY-2. Although some nodes might send $O(d(v))$ messages, we show in Section 3.2 that most nodes send (and receive) only $O(\log n)$ messages in an amortized sense. Nodes that remain undecided at the end of one iteration continue to the next iteration. It is easy to implement each iteration in a constant number of rounds.

3.2 Analysis of Algorithm 2-rulingset-msg

One *phase* of the algorithm consists of Steps 3-19, which can be implemented in a constant number of rounds. We say that a node is *decided* if it is in CATEGORY-1, CATEGORY-2, or

⁴ In the full paper [18] we present a simpler algorithm for 2-ruling set that, whp, takes $O(n^{1.5} \log n)$ messages and runs in $O(\log n)$ rounds.

⁵ Alternately, if v finds any CATEGORY-2 neighbor (that was missed by checking sampling) during broadcast step it becomes a CATEGORY-3 node and is done. However, this does not give an asymptotic improvement in the message complexity analysis compared to the stated algorithm.

Algorithm 3: Algorithm 2-rulingset-msg: code for a node v . $d(v)$ is the degree of v .

```

1 statusv = UNDECIDED;
2 while statusv = UNDECIDED do
3   if v receives a message from a CATEGORY-1 node then
4     | Set statusv = CATEGORY-2;
5   end
6   if v is UNDECIDED then it marks itself with probability  $\frac{1}{2d(v)}$  ;
7   if v is marked then
8     | ((Checking Sampling Step:) Sample a set  $A_v$  of  $4 \log(d(v))$  random neighbors
9       | independently and uniformly at random (with replacement) ;
10      | Find the categories of all nodes in  $A_v$  by communicating with them;
11      | if any node in  $A_v$  is a CATEGORY-2 node then
12        | | Set statusv = CATEGORY-3;
13      | end
14      | else
15        | | ((Local) Broadcast Step:) Send the marked status and  $d(v)$  value to all neighbors;
16        | | If v hears from an equal or higher degree (marked) neighbor then v unmarks itself;
17        | | If v remains marked, set statusv = CATEGORY-1;
18        | | Announce status to all neighbors;
19      | end
20   end
end

```

CATEGORY-3. The first lemma, which is easy to establish, shows that if a node is marked, it has a good chance to get decided.

► **Lemma 4.** *A node that marks itself in any phase gets decided with probability at least $1/2$ in that phase. Furthermore, the probability that a node remains undecided after $2 \log n$ marked phases is at most $1/n^2$.*

The next lemma bounds the round complexity of the algorithm and establishes its correctness. The round complexity bound is essentially a consequence of the previous lemma and the correctness of the algorithm is easy to check.

► **Lemma 5.** *The algorithm 2-rulingset-msg runs in $O(\Delta \log n)$ rounds whp. In particular, with probability at least $1 - 2/n^2$, a node v becomes decided after $O(d(v) \log n)$ rounds. When the algorithm terminates, i.e., when all nodes are decided, the CATEGORY-1-nodes form a 2-ruling set of the graph. Moreover, each node is correctly classified according to its category.*

We now show a technical lemma that is crucially used in proving the message complexity bounds of the algorithm in Lemma 7. It gives a high probability bound on the total number of messages sent by all nodes during the Broadcast step in any particular phase (i.e., Step 14) of the algorithm in terms of a quantity that depends on the number of undecided nodes and their neighbors. While bounding the expectation is easy, showing concentration is more involved. (We note that we really use only part (b) of the Lemma for our subsequent analysis, but showing part (a) first, helps understand the proof of part (b)).

► **Lemma 6.** *Let $U \subseteq V$ be a (sub-)set of undecided nodes at the beginning of a phase. Let $N(v)$ be the set of neighbors of v . Then the following statements hold:*

(a) *Let $Z(U) = U \cup (\cup_{v \in U} N(v))$. The total number of messages sent by all nodes in U during the Broadcast step in this phase (i.e., Step 14) of the algorithm is $O(|Z(U)| \log n)$ with probability at least $1 - 1/n^3$.*

(b) *Let $N'(v)$ be the set of undecided and category 3 neighbors of v and suppose $|N'(v)| \geq d(v)/2$ (where $d(v)$ is the degree of v), for each $v \in U$. Let $Z'(U) = U \cup (\cup_{v \in U} N'(v))$. The*

total number of messages sent by all nodes in U during the Broadcast step in this phase (i.e., Step 14) of the algorithm is $O(|Z'(U)| \log n)$ with probability at least $1 - 1/n^3$.

► **Lemma 7.** *The algorithm 2-rulingset-msg uses $O(n \log^2 n)$ messages whp.*

Proof. We will argue separately about two kinds of messages that any node can initiate. Consider any node v . **1. type 1 messages:** In the checking sampling step in some phase, v samples $4 \log d(v)$ random neighbours which costs $O(\log d(v))$ messages in that phase.

2. type 2 messages: In the broadcast step in some phase, v sends to all its neighbors which costs $d(v)$ messages. This happens when all the sampled neighbors in set A_v (found in Step 9) are not CATEGORY-2 nodes.

Note that v initiates any message at all, i.e., both type 1 and 2 messages happen, only when v marks itself, which happens with probability $1/2d(v)$.

We first bound the type 1 messages sent overall by all nodes. By the above statement, a node does checking sampling when it marks itself which happens with probability $1/2d(v)$. By Lemma 4, with probability at least $1 - 1/n^2$, a node is marked (before it gets decided) at most $2 \log n$ times. Hence, with probability at least $1 - 1/n^2$, the number of type 1 messages sent by node v is at most $O(\log d(v) \log n)$; this implies, by union bound, that with probability at least $1 - 1/n$ every node v sends at most $O(\log d(v) \log n)$ type 1 messages. Thus, whp, the total number of type 1 messages sent is $\sum_{v \in V} O(\log d(v) \log n) = O(n \log^2 n)$.

We next bound the type 2 messages, i.e., messages sent during the broadcast step. There are two cases to consider in any phase.

Case 1. In this case we focus (only) on the broadcast messages of the set U of undecided nodes v that (each) have at least $d(v)/2$ neighbors that are in CATEGORY-3 or undecided (in that phase). We show by a charging argument that any node receives amortized $O(\log n)$ messages (whp) in this case. When a node u (in this case) broadcasts, its $d(u)$ messages are charged equally to itself and its CATEGORY-3 and undecided neighbors (which number at least $d(u)/2$).

We first show that any CATEGORY-3 or undecided node v is charged by amortized $O(\log n)$ messages in any phase. Consider the set $U(v)$ which is the set of undecided nodes (each of which satisfy Case 1 property of having at least half of its neighbors that are in CATEGORY-3 or undecided in this phase) in the closed neighborhood of v (i.e., $\{v\} \cup N(v)$). Note that v will be charged only by messages broadcast by nodes in the closed neighborhood of v (this “local charging” is needed for the subsequent argument in the next para). Hence we consider the set $Z'(U(v))$ as in Lemma 6.(b), i.e., define $Z'(U(v)) = U(v) \cup (\cup_{w \in U(v)} N'(w))$, where $N'(w)$ is the set of all UNDECIDED or CATEGORY-3 neighbors of w . Since, by assumption of Case 1, every undecided node $u \in U(v)$ has at least $d(u)/2$ neighbors that are in CATEGORY-3 or UNDECIDED in the current phase, applying Lemma 6 (part (b)) to the set $Z'(U(v))$ tells us that, with probability at least $1 - 1/n^2$, the total number of messages broadcast by undecided nodes in $U(v)$ is $O(|Z'(U(v))| \log n)$. Hence, amortizing over the total number of (undecided and CATEGORY-3) nodes in $Z'(U(v))$, we show that v (which is part of $Z'(U(v))$) is charged (amortized) $O(\log n)$ in a phase. Taking a union bound, gives a high probability result for all nodes v .

To show that the same node v is not charged too many times *across phases*, we use the fact that CATEGORY-2 nodes are never charged (and they do not broadcast). We note that if a node enters the ruling set (i.e., becomes CATEGORY-1) in some phase, then all its neighbors become CATEGORY-2 nodes and will never be charged again (in any subsequent phase). Furthermore, since a marked node enters the ruling set with probability at least $1/2$, a neighbor of v (or v itself) gets charged at most $O(\log n)$ times whp. Hence overall a node

is charged at most $O(\log^2 n)$ times whp and by union bound, every node gets charged at most $O(\log^2 n)$ times whp.

Case 2. In this case, we focus on the messages broadcast by those undecided nodes v that have at most $d(v)/2 - 1$ neighbors that are in CATEGORY-3 or undecided, i.e., at least $d(v)/2 + 1$ neighbors are in CATEGORY-2. By the description of our algorithm, a node enters the broadcast step, only if checking sampling step (Step 8) fails to find a CATEGORY-2 node. The probability of this “bad” event happening is at most $\frac{1}{d(v)^4}$, which is the probability that a CATEGORY-2 neighbor (of which there are at least $d(v)/2$ many) is not among any of the $4\log(d(v))$ randomly sampled neighbors. We next bound the total number of broadcast messages generated by all undecided nodes in Case 2 during the entire course of the algorithm. By Lemma 4, for any node v , Case 2 can potentially happen only $2\log n$ times with probability at least $1 - 1/n^2$, since that is the number of times v can get marked. Let r.v. Y_v denote the number of Case 2 broadcast messages sent by v during the course of the algorithm. Conditional on the fact that it gets marked at most $2\log n$ times, we have $E[Y_v] = 2\log n \frac{1}{d(v)^4} d(v) = 2\log n \frac{1}{d(v)^3}$.

Let $Y = \sum_{v \in V} Y_v$. Hence, conditional on the fact that each node gets marked at most $2\log n$ times (which happens with probability $\geq 1 - 1/n$) the total expected number of Case 2 broadcast messages sent by all nodes is $E[Y] = \sum_{v \in V} E[Y_v] = \sum_{v \in V} 2\log n \frac{1}{d(v)^3} = O(n \log n)$.

We next show concentration of Y (conditionally as mentioned above). We know that $\text{Var}[Y_v] = 4\log^2 n \left(\frac{1}{d(v)^2} - \frac{1}{d(v)^6} \right) \leq 4\log^2 n$. Since the random variables Y_v are independent, we have $\text{Var}[Y] = \sum_{v \in V} \text{Var}(Y_v) = 4n \log^2 n$. Noting that $Y_v - E[Y_v] \leq 2n \log n$, we apply Bernstein’s inequality [9] to obtain

$$\Pr(Y \geq E[Y] + 4n \log^2 n) \leq \exp\left(-\frac{16n^2 \log^4 n}{8n \log^2 n + (2/3)2n \log n(4n \log^2 n)}\right) \leq O(1/n^2).$$

Since the conditioning with respect to the fact that all nodes get marked at most $2\log n$ times happens with probability at least $1 - 1/n$, unconditionally, $\Pr(Y \geq \Theta(n \log^2 n)) \leq O(1/n^2) + 1/n$. Hence, the overall broadcast messages sent by nodes in Case 2 is bounded by $O(n \log^2 n)$ whp.

Combining type 1 and type 2 messages, the overall number of messages is bounded by $O(n \log^2 n)$ whp. \blacktriangleleft

Thus we obtain the following theorem. In the full paper [18], we show that this analysis of the Algorithm 2-rulingset-msg is tight up to a polylogarithmic factor.

► Theorem 8. *The algorithm 2-rulingset-msg computes a 2-ruling set using $O(n \log^2 n)$ messages and terminates in $O(\Delta \log n)$ rounds with high probability.*

4 Message Complexity Lower Bounds

We first point out that the bound of Theorem 8 is tight up to logarithmic factors. The proof is a simple indistinguishability argument and is relegated to the full paper [18].

► Theorem 9. *Any $O(1)$ -ruling set algorithm that succeeds with probability $1 - o(1)$ sends $\Omega(n)$ messages in the worst case. This is true even if nodes have prior knowledge of the network size n .*

Next, we show a separation between the message complexity of computing an β -ruling set ($\beta > 1$) and an MIS (i.e., 1-ruling set) by proving an unconditional $\Omega(n^2)$ lower bound for the latter.

► **Theorem 10.** *Any maximal independent set algorithm that succeeds with probability $1 - \epsilon$ on connected networks, where $0 \leq \epsilon < \frac{1}{2}$ is a constant, has a message complexity of $\Omega(n^2)$ in expectation. This is true even if nodes have prior knowledge of the network size n .*

Proof. For the sake of a contradiction, assume that there is an algorithm \mathcal{A} that, with probability $1 - o(1)$, sends at most $\mu = o(n^2)$ messages. Moreover, assume that \mathcal{A} succeeds with probability $\geq 1 - \epsilon$, for some $\epsilon < \frac{1}{2}$. In the remainder of the proof, we condition on \mathcal{A} sending at most $o(n^2)$ messages.

Consider two copies G and G' of the complete bipartite graph on $n/2$ nodes.⁶ For now, we consider the anonymous case where nodes do not have access to unique IDs; we will later show how to remove this restriction. Recall that in our model (cf. Section 1.1), we assume that nodes do not have any prior knowledge of their neighbors in the graph. Instead, each node u has a list ports $1, \dots, deg_u$, whose destination are wired in advance by an adversary.

We consider two concrete instances of our lower bound network depending on the wiring of the edges. First, let $D = (G, G')$ be the disconnected graph consisting of G and G' and their induced edge sets. It is easy to see that there are exactly 4 possible choices for an MIS on D , as any valid MIS must contain the entire left (resp. right) half of the nodes in G and G' and no other nodes. We denote the events of obtaining one of the four possible MISs by LL', LR', RL', RR' , where, e.g., RL' is the event that the right half of G (i.e. nodes in R) and the left half of G' (i.e. nodes in L') are chosen. Let “on D ” be the event that \mathcal{A} is executed on graph D . Of course, we cannot assume that algorithm \mathcal{A} does anything useful on this graph as we require \mathcal{A} only to succeed on connected networks. However, we will make use of the symmetry of the components of D later on in the proof.

► **Observation 1.** $\Pr[LL' \mid \text{on } D] = \Pr[LR' \mid \text{on } D] = \Pr[RL' \mid \text{on } D] = \Pr[RR' \mid \text{on } D]$.

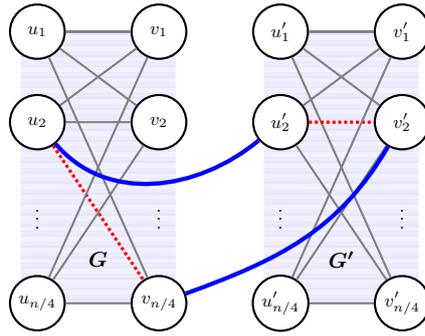
We now define the second instance of our lower bound graph. Consider any pair of edges $e = (u, v) \in G = (L, R)$ and $e' = (u', v') \in G' = (L', R')$. We define the *bridge graph* by removing e and e' from G respectively G' and, instead, adding the *bridge edges* $b = (u, u')$ and $b' = (v, v')$ by connecting the same ports that were used for e and e' ; see Figure 1. We use B to denote a graph that is chosen uniformly at random from all possible bridge graphs, i.e., the edges replaced by bridge edges are chosen uniformly at random according to the above construction. Let “ $G \leftrightarrow G'$ ” be the event that \mathcal{A} sends at least 1 message over a bridge edge and, similarly, we use “ $G \not\leftrightarrow G'$ ” to denote the event that this does not happen.

► **Lemma 11.** *Consider an execution of algorithm \mathcal{A} on a uniformly at random chosen bridge graph B . The probability that a message is sent across a bridge is $o(1)$, i.e., $\Pr[G \not\leftrightarrow G'] = 1 - o(1)$.*

A crucial property of our construction is that, as long as no bridge edge is discovered, the algorithm behaves the same on B as it does on D . The following lemma can be shown by induction over the number of rounds.

► **Lemma 12.** *Let Y be any event that is a function of the communication and computation performed by algorithm \mathcal{A} . Then, $\Pr[Y \mid G \not\leftrightarrow G'] = \Pr[Y \mid \text{on } D]$.*

⁶ To simplify our analysis, we assume that $n/2$ and $n/4$ are integers.



■ **Figure 1** The lower bound graph $B(G, G')$ for Theorem 9 with bridge edges (u_2, u'_2) and $(v_{n/4}, v'_{n/4})$. The disconnected graph D is given by replacing the bridge edges with the dashed edges.

Now consider a run of algorithm \mathcal{A} on a uniformly at random chosen bridge graph B . Let “ \mathcal{A} succ.” denote the event that \mathcal{A} correctly outputs a MIS. Observe that \mathcal{A} succeeds when executed on B if and only if we arrive at an output configuration corresponding LR' or RL' . It follows that $\Pr[\mathcal{A} \text{ succ.}] = \sum_{W \in \{LR', RL'\}} \Pr[W \mid G \not\leftrightarrow G'] \cdot \Pr[G \not\leftrightarrow G'] + \Pr[\mathcal{A} \text{ succ.} \mid G \leftrightarrow G'] \cdot \Pr[G \leftrightarrow G'] \geq 1 - \epsilon$. Lemma 11 tells us that $\Pr[G \leftrightarrow G'] = o(1)$ and, using $\Pr[G \not\leftrightarrow G'] \leq 1$, allows us to rewrite the above inequality as $\sum_{W \in \{LR', RL'\}} \Pr[W \mid G \not\leftrightarrow G'] \geq 1 - \epsilon - o(1)$. Applying Lemma 12 to the terms in the sum, we get $\sum_{W \in \{LR', RL'\}} \Pr[W \mid \text{on } D] \geq 1 - \epsilon - o(1)$. By Observation 1, we know that $\Pr[LR' \mid \text{on } D] + \Pr[RL \mid \text{on } D] \leq \frac{1}{2}$, which we can plug into the previously obtained bound on $\sum_{W \in \{LR', RL'\}} \Pr[W \mid \text{on } D]$ to obtain $\epsilon \geq \frac{1}{2} - o(1)$, yielding a contradiction to ϵ being a constant less than $\frac{1}{2}$. So far, we have conditioned on \mathcal{A} sending at most $o(n^2)$ messages. Since this event happens with probability $1 - o(1)$, removing the conditioning reduces the above bound on ϵ by at most $o(1)$, which still provides a contradiction.

Finally, we can remove the restriction of not having unique IDs by arguing that the algorithm can generate unique IDs with high probability, since we assume that nodes know n ; see the proof of Theorem 9 in the full paper [18] for a similar argument. This completes the proof of Theorem 10.

5 Conclusion

Several key open questions are motivated by the results in this paper. First, can the MIS lower bounds in the LOCAL model shown by Kuhn et al. [15] be extended to 2-ruling sets? In an orthogonal direction, can we derive time lower bounds for MIS in the CONGEST model, that are stronger than their LOCAL-model counterparts? And on the algorithms side, can we improve ruling-set time bounds in the CONGEST model. e.g., by showing that the 2-ruling set problem can be solved in $O(\log^\alpha n)$ rounds, $\alpha < 1$, in CONGEST?

Second, although we have presented near-tight message bounds for 2-ruling sets, we don't have a good understanding of the message-time *tradeoffs*. In particular, a key question is whether we can design a 2-ruling set algorithm that uses $O(n \text{ polylog } n)$ messages, while running in $O(\text{polylog } n)$ rounds? More generally, can we obtain a tradeoff that characterizes the dependence of one measure on the other or obtain lower bounds on the complexity of one measure while fixing the other measure.

References

- 1 Noga Alon, László Babai, and Alon Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *J. Algorithms*, 7(4):567–583, 1986.
- 2 Baruch Awerbuch, Oded Goldreich, David Peleg, and Ronen Vainish. A trade-off between information and communication in broadcast protocols. *J. ACM*, 37(2):238–256, 1990.
- 3 Leonid Barenboim, Michael Elkin, and Fabian Kuhn. Distributed $(\delta + 1)$ -coloring in linear (in δ) time. *SIAM Journal on Computing*, 43(1):72–95, 2014.
- 4 Leonid Barenboim, Michael Elkin, Seth Pettie, and Johannes Schneider. The locality of distributed symmetry breaking. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 321–330, 2012.
- 5 Leonid Barenboim, Michael Elkin, Seth Pettie, and Johannes Schneider. The locality of distributed symmetry breaking. *J. ACM*, 63(3):20:1–20:45, June 2016.
- 6 József Beck. An algorithmic approach to the Lovász Local Lemma. I. *Random Struct. Algorithms*, 2(4):343–365, December 1991.
- 7 Tushar Bisht, Kishore Kothapalli, and Sriram Pemmaraju. Brief announcement: Super-fast t-ruling sets. In *Proceedings of the 2014 ACM Symposium on Principles of Distributed Computing, PODC '14*, pages 379–381, New York, NY, USA, 2014. ACM.
- 8 T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*. The MIT Press, 1990.
- 9 Devdatt Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, New York, NY, USA, 1st edition, 2009.
- 10 Mohsen Ghaffari. An improved distributed algorithm for maximal independent set. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 270–277, 2016.
- 11 A. Goldberg, S. Plotkin, and G. Shannon. Parallel symmetry breaking in sparse graphs. *SIAM J. Discrete Math.*, 1:434–446, 1989.
- 12 Monika Henzinger, Sebastian Krininger, and Danupon Nanongkai. A deterministic almost-tight distributed algorithm for approximating single-source shortest paths. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 489–498, 2016.
- 13 Hartmut Klauck, Danupon Nanongkai, Gopal Pandurangan, and Peter Robinson. Distributed computation of large-scale graph problems. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 391–410, 2015.
- 14 Kishore Kothapalli and Sriram V. Pemmaraju. Super-fast 3-ruling sets. In *FSTTCS*, pages 136–147, 2012.
- 15 Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. Local computation: Lower and upper bounds. *J. ACM*, 63(2):17:1–17:44, March 2016.
- 16 Shay Kutten, Gopal Pandurangan, David Peleg, Peter Robinson, and Amitabh Trehan. On the complexity of universal leader election. *J. ACM*, 62(1):7:1–7:27, 2015. doi:10.1145/2699440.
- 17 M. Luby. A simple parallel algorithm for the maximal independent set. *SIAM Journal on Computing*, 15:1036–1053, 1986.
- 18 Shreyas Pai, Gopal Pandurangan, Sriram V. Pemmaraju, Talal Riaz, and Peter Robinson. Symmetry Breaking in the Congest Model: Time- and Message-Efficient Algorithms for Ruling Sets. *ArXiv e-prints*, May 2017.
- 19 A. Panconesi and A. Srinivasan. On the complexity of distributed network decomposition. *J. Algorithms*, 20(2):356–374, 1996.
- 20 David Peleg. *Distributed computing: a locality-sensitive approach*. Society for Industrial and Applied Mathematics, 2000.