Topology recognition with advice[*]

Emanuele G. Fusco[1], Andrzej Pelc[†2], and Rossella Petreschi[3]

[1]Department of Computer, Control, and Management Engineering
Antonio Ruberti, Sapienza, University of Rome, 00185 Rome,
Italy. `fusco@diag.uniroma1.it`.
[2] Département d'informatique, Université du Québec en
Outaouais, Gatineau, Québec J8X 3X7, Canada. `pelc@uqo.ca`.
[3]Computer Science Department, Sapienza, University of Rome,
00198 Rome, Italy. `petreschi@di.uniroma1.it`.

Abstract

In topology recognition, each node of an anonymous network has to deterministically produce an isomorphic copy of the underlying graph, with all ports correctly marked. This task is usually unfeasible without any a priori information. Such information can be provided to nodes as advice. An oracle knowing the network can give a (possibly different) string of bits to each node, and all nodes must reconstruct the network using this advice, after a given number of rounds of communication. During each round each node can exchange arbitrary messages with all its neighbors and perform arbitrary local computations. The time of completing topology recognition is the number of rounds it takes, and the size of advice is the maximum length of a string given to nodes.

We investigate tradeoffs between the time in which topology recognition is accomplished and the minimum size of advice that has to be given to nodes. We provide upper and lower bounds on the minimum size of advice that is sufficient to perform topology recognition in a given time, in the class of all graphs of size n and diameter $D \leq \alpha n$, for any constant $\alpha < 1$. In most cases, our bounds are asymptotically tight.

Keywords
topology recognition, network, advice, time, tradeoff

1   Introduction

1.1   The model and the problem

Learning an unknown network by its nodes is one of the fundamental distributed tasks in networks. Once nodes acquire a faithful labeled map of the network, any other distributed task, such as leader election [21, 28], minimum weight spanning tree construction [5], renaming [4], etc. can be performed by nodes using only local computations. Knowing topology also simplifies the task of routing. More generally, constructing a labeled map converts all distributed problems to centralized ones, in the sense that nodes can solve them simulating a central monitor.

If nodes are a priori equipped with unique identifiers, they can deterministically construct a labeled map of the network, by exchanging messages, without any additional information about the network. However, even if nodes have unique identities, relying on them for the task of learning the network is not always possible. Indeed, nodes may be reluctant to reveal their identities for security or privacy reasons. Hence it is important to design algorithms reconstructing the topology of the network without assuming any node labels, i.e., for anonymous networks. In this paper we are interested in deterministic solutions.

Ports at each node of degree d are arbitrarily numbered $0, \ldots, d-1$, and there is no assumed coherence between port numbers at different nodes. A node is aware of its degree, and it knows on which port it sends or receives a message. The goal is, for each node, to get an isomorphic copy of the graph underlying the network, with all port numbers correctly marked. (Port numbers are very useful, e.g., to perform the routing task.) There are two variants of this task: a weaker version, that we call *anonymous topology recognition*, in which the nodes of the reconstructed map are unlabeled, and a stronger version, that we call *labeled topology recognition*, in which all nodes construct a map of the network assigning distinct labels to all nodes in the same way, and know their position in this map. Even anonymous topology recognition is not always feasible without any a priori information given to nodes, as witnessed, e.g., by the class of oriented rings in which ports at each node are numbered 0,1 in clockwise order. No amount of information exchange can help nodes to recognize the size of the oriented ring and hence to reconstruct correctly its topology. Thus, in order to accomplish (even anonymous) topology recognition for arbitrary networks, some information must be provided to nodes. This can be done in the form of *advice*. An oracle knowing the network gives a (possibly different) string of bits to each node. Then nodes execute a deterministic distributed algorithm that does not assume knowledge of the network, but uses message exchange and the advice provided by the oracle to nodes, in order to reconstruct the topology of the network by each of its nodes.

In this paper we study tradeoffs between the *size of advice* provided to nodes and the *time* of topology recognition. The size of advice is defined as the length of the longest string of bits given by the oracle to nodes. For communication, we use the extensively studied $\mathcal{LOCAL}$ model [27]. In this model, communication

Time	Optimal Size of Advice
$2D + 1$	1 [Proposition 3.1]
$D + k$, for $D/2 < k \le D$ $\quad\star$	Upper B.: $O(1 + (\log n)/k)$, Lower B.: 1 [Theorem 4.1]
$D + k$, for $0 < k \le D/2$	$\Theta(1 + (\log n)/k)$ [Corollary 4.1]
D	$\Theta(n \log n)$ [Corollary 5.2]
$D - k$, for $0 < k \le D$	$\Theta((n^2 \log n)/(D - k + 1))$ [Corollary 6.1]

Table 1: The summary of results. All our bounds are tight, except for those in the line with the $\star$ when $D \in o(\log n)$.

proceeds in synchronous rounds and all nodes start simultaneously. In each round each node can exchange arbitrary messages with all its neighbors and perform arbitrary local computations. The time of completing a task is the number of rounds it takes. The central question of the paper is:

> What is the minimum size of advice that enables (anonymous or labeled) topology recognition in a given time T, in the class of n-node networks of diameter D?

It should be stressed that the reason why we adopt the $\mathcal{LOCAL}$ model is to focus on how deeply nodes should probe the network in order to discover the topology. Since nodes are anonymous, this depth of probing may often go beyond the diameter of the network, enabling the nodes to "see" other nodes several times, along different paths, and deduce some information from these deeper views. At the same time, as it is always done when using the $\mathcal{LOCAL}$ model [27], the size of messages is ignored.

1.2   Our results

We provide upper and lower bounds on the minimum size of advice sufficient to perform topology recognition in a given time, in the class $\mathcal{C}(n, D)$ of all graphs of size n and diameter $D \le \alpha n$, for any constant $\alpha < 1$. All our upper bounds are valid even for the harder task of labeled topology recognition, while our lower bounds also apply to the easier task of anonymous topology recognition. Hence we will only use the term *topology recognition* for all our results. We prove upper bounds $f(n, D, T)$ on the minimum size of advice sufficient to perform

topology recognition in a given time T, for the class $\mathcal{C}(n, D)$, by providing an assignment of advice of size $f(n, D, T)$ and an algorithm, using this advice, that accomplishes this task, within time T, for any network in $\mathcal{C}(n, D)$. We prove lower bounds on the minimum size of advice, sufficient for a given time T, by constructing graphs in $\mathcal{C}(n, D)$ for which topology recognition within this time is impossible with advice of a smaller size. (Notice that, while our upper bounds would hold even if $D \approx n$, our lower bound constructions rely on the constraint $D \leq \alpha n$.)

The meaningful span of possible times for topology recognition is between 0 and $2D+1$. Indeed, while advice of size $O(n^2 \log n)$ permits topology recognition in time 0 (i.e., without communication), we show that topology recognition in time $2D + 1$ can be done with advice of size 1, which is optimal.

For most values of the allotted time, our bounds are asymptotically tight. This should be compared to many results from the literature on the advice paradigm (see, e.g., [9, 14, 15, 17, 26]), which often either consider the size of advice needed for feasibility of a given task, or only give isolated points in the curve of tradeoffs between resources (such as time) and the size of advice.

We show that, if the allotted time is $D - k$, where $0 < k \leq D$, then the optimal size of advice is $\Theta((n^2 \log n)/(D - k + 1))$. If the allotted time is D, then this optimal size is $\Theta(n \log n)$. If the allotted time is $D + k$, where $0 < k \leq D/2$, then the optimal size of advice is $\Theta(1 + (\log n)/k)$. The only remaining gap between our bounds is for time $D + k$, where $D/2 < k \leq D$. In this time interval our upper bound remains $O(1 + (\log n)/k)$, while the lower bound (that holds for any time) is 1. This leaves a gap if $D \in o(\log n)$. See Table 1 for a summary of our results.

Our results show how sensitive is the minimum size of advice to the time allowed for topology recognition: allowing just one round more, from D to $D+1$, decreases exponentially the advice needed to accomplish this task. Our tight bounds on the minimum size of advice also show a somewhat surprising fact that the amount of information that nodes need to reconstruct a labeled map of the network, in a given time, and that needed to reconstruct an anonymous map of the network in this time, are asymptotically the same in most cases.

1.3   Related work

Many papers [1, 7, 9, 10, 11, 14, 15, 16, 17, 18, 20, 22, 23, 24, 26, 30] considered the problem of increasing the efficiency of network tasks by providing nodes with some information of arbitrary kind. This approach was referred to as algorithms using *informative labeling schemes*, or equivalently, algorithms with *advice*. Advice is given either to nodes of the network or to mobile agents performing some network task. Several authors studied the minimum size of advice required to solve the respective network problem in an efficient way. Thus the framework of advice permits to quantify the amount of information needed for an efficient solution of a given network problem, regardless of the type of information that is provided.

In [9] the authors investigated the minimum size of advice that has to be

given to nodes to permit graph exploration by a robot. In [24], given a distributed representation of a solution for a problem, the authors investigated the number of bits of communication needed to verify the legality of the represented solution. In [15] the authors compared the minimum size of advice required to solve two information dissemination problems using a linear number of messages. In [16] the authors established the size of advice needed to break competitive ratio 2 of an exploration algorithm in trees. In [17] it was shown that advice of constant size permits to carry on the distributed construction of a minimum spanning tree in logarithmic time. In [11] the advice paradigm was used for online problems. In [14] the authors established lower bounds on the size of advice needed to beat time $\Theta(\log^* n)$ for 3-coloring of a cycle and to achieve time $\Theta(\log^* n)$ for 3-coloring of unoriented trees. In the case of [26] the issue was not efficiency but feasibility: it was shown that $\Theta(n \log n)$ is the minimum size of advice required to perform monotone connected graph clearing. In [22] the authors studied radio networks for which it is possible to perform centralized broadcasting in constant time. They proved that $O(n)$ bits of advice allow to obtain constant time in such networks, while $o(n)$ bits are not enough.

Distributed computation on anonymous networks has been investigated by many authors, e.g., [2, 3, 6, 12, 13, 19, 25, 27, 31] for problems ranging from leader election to computing boolean functions and communication in wireless networks. In [12] the authors compared randomized and deterministic algorithms to solve distributed problems in anonymous networks. In [13] a hierarchy of distributed problems in anonymous networks was studied. Feasibility of topology recognition for anonymous graphs with adversarial port labelings was studied in [31]. The problem of efficiency of map construction by a mobile agent, equipped with a token, exploring an anonymous graph has been studied in [8]. In [10] the authors investigated the minimum size of advice that has to be given to a mobile agent, in order to enable it to reconstruct the topology of an anonymous network or to construct its spanning tree. Notice that the mobile agent scenario makes the problem of map construction much different from our setting. Since all the advice is given to a single agent, breaking symmetry may be very hard. Even anonymous map construction often requires providing a large amount of information to the agent, regardless of the exploration time. To the best of our knowledge, tradeoffs between time and the size of advice for topology recognition have never been studied before.

2   Preliminaries

Unless otherwise stated, we use the word *graph* to mean a simple undirected connected graph without node labels, and with ports at each node of degree d labeled $\{0, \ldots, d-1\}$. Two graphs $G = (V, E)$ and $G' = (V', E')$ are *isomorphic*, if and only if, there exists a bijection $f : V \longrightarrow V'$ such that the edge $\{u, v\}$, with port numbers p at u and q at v is in E, if and only if, the edge $\{f(u), f(v)\}$ with port numbers p at $f(u)$ and q at $f(v)$ is in E'.

The size of a graph is the number of its nodes. Throughout the paper we

consider a fixed positive constant $\alpha < 1$ and the class of graphs of size n and diameter $D \leq \alpha n$. We use log to denote the logarithm to the base 2. For a graph G, a node u in G, and any integer t, we denote by $N_t(u)$ the set of nodes in G at distance at most t from u.

We will use the following notion from [31]. The *view* from node u in graph G is the infinite tree $\mathcal{V}(u)$ rooted at u with unlabeled nodes and labeled ports, whose branches are infinite sequences of port numbers coding all infinite paths in the graph, starting from node u. The *truncated view* $\mathcal{V}^l(u)$ is the truncation of this tree to depth $l \geq 0$. to level l, for each l.

Given a graph $G = (V, E)$, a function $f : V \longrightarrow \{0, 1\}^*$ is called a *decoration* of G. Notice that an assignment of advice to nodes of G is a decoration of G. For a given decoration f of a graph G we define the *decorated graph* G_f as follows. Nodes of G_f are ordered pairs $(v, f(v))$, for all nodes v in V. G_f has an edge $\{(u, f(u)), (v, f(v))\}$ with port numbers p at $(u, f(u))$ and q at $(v, f(v))$, if and only if, E contains the edge $\{u, v\}$, with port numbers p at u and q at v.

We define the *decorated view* at depth l of node v in G, according to f, as the truncated view at depth l of node $(v, f(v))$ in the decorated graph G_f. Nodes in the decorated view are labeled with the values assigned to the corresponding nodes of G by the decorating function f.

The following two lemmas will be used in the proofs of our upper bounds.

Lemma 2.1 *Let G be a graph and let r be a positive integer. There exists a set X of nodes in G satisfying the following conditions.*

- *For any node w of G there exists a node u in X such that the distance between w and u is at most r.*

- *For each pair $\{u, v\}$ of distinct nodes in X, the distance between u and v is larger than r.*

Proof: The set proving the lemma is any independent dominating set of the rth power of the graph G. $\square$

Lemma 2.2 *Let G be a graph of diameter D and let A be an injective decoration of G. Then each node u in G can accomplish topology recognition using its view, decorated according to A, at depth $D + 1$, even without knowing D a priori.*

Proof: Let l be the minimum integer such that the views at depth l and $l + 1$, decorated according to A, contain the same set of values. Then the view at depth $l + 1$ contains all nodes and all edges of the graph G, and thus it is sufficient to reconstruct the graph G and its decoration A. The lemma follows from the fact that the view at depth D contains all nodes of the graph. $\square$

The following proposition can be easily proved by induction on the round number. Intuitively it says that, if two nodes executing the same algorithm have the same decorated views at depth t, then they behave identically for at least t rounds.

Proposition 2.1 *Let G and G' be two graphs, let u be a node of G and let u' be a node of G'. Let A be a decoration of G and let A' be a decoration of G'. Let $\mathcal{A}$ be any topology recognition algorithm. Let σ_t be the set of triples $\langle p, r, m \rangle$, where m is the message received by node u in round $r \leq t$ through port p when executing algorithm $\mathcal{A}$ on the graph G, decorated according to A. Let σ'_t be defined as σ_t, but for u', G', and A' instead of u, G, and A.*

If the view of u at depth t, decorated according to A is the same as the view of u' at depth t, decorated according to A', then $\sigma_t = \sigma'_t$.

We will use the above proposition to prove our lower bounds as follows. If the size of advice is too small, then there are two non-isomorphic graphs G and G' resulting, for some node u in G and some node u' in G', in the same decorated view at the depth equal to the time available to perform topology recognition. Hence either u or u' must incorrectly reconstruct the topology (even anonymous) of G or G'.

3   Time $2D + 1$

We start our analysis by constructing a topology recognition algorithm that works in time $2D + 1$ and uses advice of size 1. Since we will show that, for arbitrary $D \geq 3$, there are networks in which topology recognition without advice is impossible in any time, this shows that the meaningful time-span to consider for topology recognition is between 0 and $2D + 1$.

Algorithm TR-1
Advice:
The oracle assigns bit 1 to one node (call it v), and bit 0 to all others. Let A be this assignment of advice.
Node protocol:
In round i, each node u sends its view at depth $i-1$, decorated according to A, to all its neighbors; it receives such views from all its neighbors and constructs its view at depth i, decorated according to A. This task continues until termination of the algorithm.

Let t be the smallest round number at which node u sees a node with advice 1 in its view decorated according to A (at depth t). Node u assigns to itself a label in round t as follows. The label $\ell(u)$ is the lexicographically smallest shortest path, defined as a sequence of consecutive port numbers (each traversed edge corresponds to a pair of port numbers), from u to any node with advice 1, in its decorated view at depth t. (Notice that since there can be many shortest paths between u and v, this node can appear many times in the decorated view at depth t of u.) Let A^* be the decoration corresponding to the labeling obtained as above. (We will show that labels in A^* are unique.)

After round t, node u starts constructing its decorated view, according to decoration A^*. In any round $t' > t$, node u sends both its view, decorated according to A, at depth t', and its view, decorated according to A^*, at the largest possible depth. Messages required to perform this task are piggybacked

7

to those used for constructing views, decorated according to A, at increasing depths. In each round t', node u checks for newly discovered values of A^*. As soon as there are no new values, node u reconstructs the labeled map and outputs it. Then node u computes the diameter D of the resulting graph and continues to send its views, decorated according to A and according to A^*, at increasing depths, until round $2D + 1$. After round $2D + 1$ node u terminates. ◇

Proposition 3.1 *Algorithm* TR-1 *completes topology recognition for all graphs of size n and diameter D in time $2D + 1$, using advice of size 1.*

Proof: We first prove that the topology reconstructed at each node is correct.

Uniqueness of labels assigned according to A^* follows from the fact that the node v with advice 1 is unique and, for each node u, the lexicographically smallest shortest path between u and v is unique. Since, in a graph of diameter D, every node is at distance at most D from v, each node u computes its label $\ell(u)$ within D rounds.

Notice that some nodes will acquire their unique label earlier than others. However, since each node acquires its complete decorated view (according to A^*) at depth i within i rounds after all nodes at distance at most i from it have acquired their unique label, after $2D + 1$ rounds of communication all nodes have their decorated view (according to A^*) at depth $D + 1$. By making each node continue to participate in the construction of decorated views at increasing depths until round $2D + 1$ (even after it reconstructed the topology), the algorithm guarantees that *all* nodes will be able to get their views, decorated according to A^*, at depth $D + 1$. Correctness of the reconstructed topology follows from Lemma 2.2. Since all nodes stop in round $2D + 1$ and the advice provided to each node consists of 1 bit, the proof is complete. □

The following proposition, cf. [31], shows that advice of size 1, as used by Algorithm TR-1, is necessary, regardless of the allotted time. As opposed to the n-node rings mentioned in the introduction as graphs that require at least one bit of advice, but whose diameter is $\lfloor n/2 \rfloor$, the class of graphs we will use to prove the proposition allows greater flexibility of the diameter.

Proposition 3.2 *Let $D \geq 3$ and let $n \geq D + 6$ be an even integer. The size of advice needed to perform topology recognition for the class of all graphs of size n and diameter D is at least 1.*

Proof: Consider the two graphs G_1 and G_2 in Figure 1.a that where constructed in [31]. They have the following properties:

- they are non-isomorphic;

- both of them have size 6 and diameter 3;

- all black nodes in both graphs have the same view;

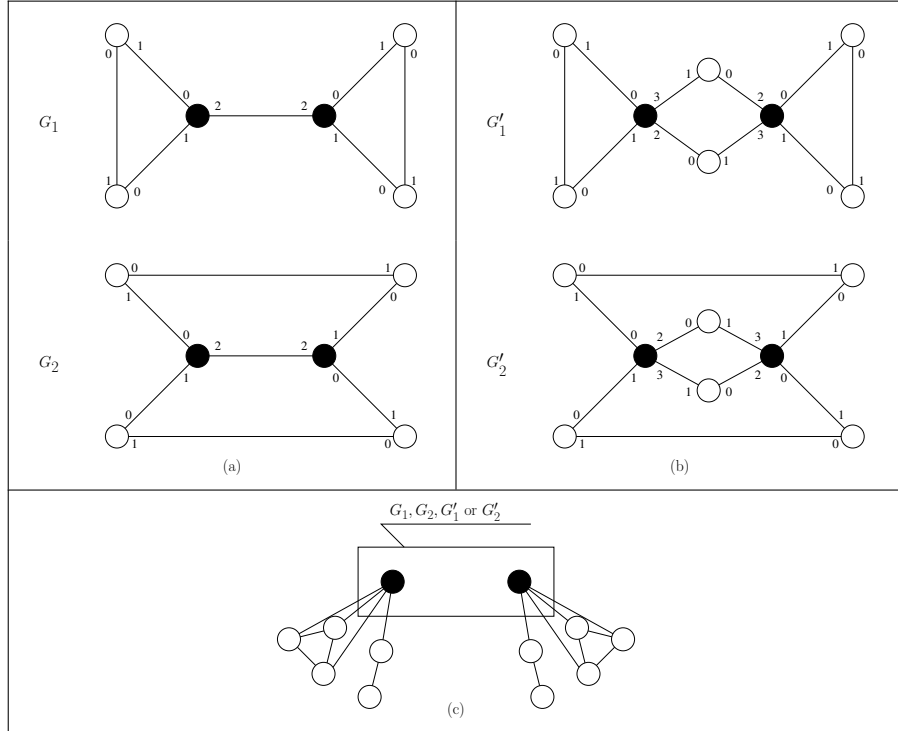- the black nodes are adjacent in both graphs.

Figure 1: a) Graphs G_1 and G_2 from [31]. b) Graphs G_1' and G_2'. c) Schematic representation of graphs H_1 and H_2, in which the rectangle represents one of the graphs G_1, G_2, G_1', or G_2'.

The graphs G'_1 and G'_2 in Figure 1.b are obtained from G_1 and G_2, respectively, replacing the edge connecting the two black nodes by two paths of length 2. These graphs have the following properties:

- they are non-isomorphic;

- both of them have size 8 and diameter at most 4;

- all black nodes in both graphs have the same view;

- the black nodes are at distance 2 in both graphs.

For $D \geq 3$ and even $n \geq D+6$ we construct two graphs H_1 and H_2, of size n and diameter D, as follows. If D is odd and $D = 2k+1$, we obtain H_1 by connecting two identical copies of a path of length k and of a clique of size $(n - 2k - 6)/2$ to each black node of the graph G_1, in a symmetric fashion. We obtain H_2 by doing the same for graph G_2. If D is even and $D = 2k + 2$, we obtain H_1 by connecting two identical copies of a path of length k and of a clique of size $(n - 2k - 8)/2$ to each black node of the graph G'_1, in a symmetric fashion. We obtain H_2 by doing the same for graph G'_2. (See Figure 1.c.)

Notice that graphs H_1 and H_2 have diameter D and size n, they are non-isomorphic both for D odd and for D even, and all black nodes in both graphs have the same view. Hence, in view of Proposition 2.1, black nodes cannot correctly perform (even anonymous) topology recognition in H_1 or in H_2 without any advice. $\qquad\square$

4 Time above D

In this section we study the size of advice sufficient to perform topology recognition in arbitrary time larger than D (but smaller than $2D + 1$), i.e., large enough for allowing each node to see all nodes and edges of the graph. We first give an algorithm using advice of size $O(1 + \log(n)/k)$ that performs topology recognition in time $D + k$.

Algorithm TR-2
Advice:
Let G be a graph of size n and diameter D. Let $t = \lceil k/4 \rceil - 1$.

If $t = 0$ then the oracle gives a unique label of size $\lceil \log n \rceil$ as advice to each node.

Suppose that $t \geq 1$. The oracle picks a set of nodes X satisfying Lemma 2.1, for $r = 2t$. Then it chooses a unique label $\ell(v)$ from the set $\{0, \ldots, n - 1\}$ for each node v in X. For any node $u \in N_{t-1}(v)$ let $\pi_v(u)$ be the lexicographically smallest shortest path (coded as a sequence of consecutive port numbers) from u to v. Sort the nodes u in $N_{t-1}(v)$ in the increasing lexicographic order of $\pi_v(u)$. The binary representation of $\ell(v)$ is partitioned into $|N_{t-1}(v)|$ consecutive segments, each of length at most $\lceil (\log n)/|N_{t-1}(v)| \rceil$. The oracle assigns the first segment, with a trailing bit 1, as advice to node v. For $1 < i \leq |N_{t-1}(v)|$,

the i-th segment, with a trailing bit 0, is assigned as advice to the i-th node of $N_{t-1}(v)$. (Notice that some nodes in $N_{t-1}(v)$ could receive only the trailing bit 0 as advice.) All other nodes get the empty string as advice. Let A_1 be the above assignment of advice.

Node protocol:

We first describe the protocol when $t \geq 1$. In round i, each node u sends its view at depth $i - 1$, decorated according to A_1, to all its neighbors; it receives such views from all its neighbors and constructs its view at depth i, decorated according to A_1. This task continues until termination of the algorithm.

Each node u whose advice has a trailing bit 0 assigns to itself a temporary label $\ell'(u)$ as follows. Let s be the smallest round number at which node u sees a node with advice with a trailing bit 1 in its view decorated according to A_1 (at depth s). The label $\ell'(u)$ is the lexicographically smallest shortest path, defined as a sequence of consecutive port numbers, from u to any node with advice with a trailing bit 1 in its view, decorated according to A_1, at depth s.

Let u be a node whose advice has a trailing bit 0. After reconstructing its label $\ell'(u)$, node u sends $(\ell'(u), A_1(u))$ to the node $v \in X$ closest to it, along the lexicographically smallest shortest path that determined label $\ell'(u)$. Nodes along this path relay these messages piggybacking them to any message that they should send in a given round.

Each node $v \in X$ (having a trailing bit 1 in its advice) computes t as the first depth in which its view, decorated according to A_1 contains nodes without any advice. In round $2t$ each such node reconstructs its label $\ell(v)$ from messages $(\ell'(u), A_1(u))$ it received (which it sorts in the increasing lexicographic order of $\ell'(u)$), and from $A_1(v)$.

Let A_2 be the decoration of G where each node v in the set X is mapped to the binary representation of its unique label $\ell(v)$, and each node outside of X is mapped to the empty string.

Nodes outside of X start constructing their decorated view, according to A_2. This construction is put on hold by a node v in X until the time when it reconstructs its unique label $\ell(v)$. Upon reconstructing its label $\ell(v)$, each node $v \in X$ starts constructing its view decorated according to A_2, hence allowing its neighbors to construct their view, decorated according to A_2, at depth 1. This process continues for $2t$ steps, during which nodes construct and send their views at increasing depth, decorated according to A_2.

Each node u assigns a label $\ell''(u)$ to itself as follows. Let s' be the smallest depth at which the view of u, decorated according to A_2, contains a node v with label $\ell(v)$ and let $\lambda(u, v)$ be the lexicographically smallest path connecting u to such a node v (coded as a sequence of consecutive port numbers). Node u sets $\ell''(u) = (\lambda(u, v), \ell(v))$.

Let A_3 be the decoration of G where each node u is mapped to $\ell''(u)$. (We will prove that A_3 is an injective function.) Upon computing its value in A_3 each node starts constructing its decorated view, according to A_3. In each round t', node u checks for newly discovered values of A_3. As soon as there are no new values, node u reconstructs the labeled map and outputs it. Then node u computes the diameter D of the resulting graph and continues to send its views,

decorated according to A_1, according to A_2, and according to A_3, at increasing depths, until round $D + 4t + 1$. After round $D + 4t + 1$ node u terminates.

If $t = 0$, the protocol consists only of the last step described above, with decoration A_3 replaced by the assignment of advice given to nodes by the oracle. ◇

Theorem 4.1 *Let $0 < k \leq D$. Algorithm* TR-2 *completes topology recognition for all graphs of size n and diameter D within time $D + k$, using advice of size $O(1 + (\log n)/k)$.*

Proof: Let G be a graph of size n and diameter D. If $t = \lceil k/4 \rceil - 1 = 0$, then k is constant and each node receives advice of size $\lceil \log n \rceil$. In this case the proof follows from Lemma 2.2. Hence in the sequel we can assume that $t \geq 1$.

Let X be the set of nodes satisfying Lemma 2.1, for $r = 2t$, selected by the oracle. Due to the properties of the set X, the sets $N_t(v)$ and consequently the sets $N_{t-1}(v)$ are pairwise disjoint for distinct nodes $v \in X$. Hence, no node would receive more than one segment of a label $\ell(v)$, for some node $v \in X$.

Within round $2t - 2$, each node $v \in X$ receives the complete set of segments $A_1(u)$, for $u \in N_{t-1}(v)$, of its label $\ell(v)$. Moreover, it can reconstruct the order of the segments according to the lexicographic order of temporary labels $\ell'(u)$ received together with the corresponding segments. Hence, by round $2t$, all nodes in X know their unique label $\ell(v)$.

Since each node of G is at distance at most $2t$ from some node in X and A_2 is an injective function for nodes in X, the view decorated according to A_2 at depth $2t$ of each node contains some uniquely labeled node. Consequently, $A_3(u)$ can be computed by each node u within $4t$ rounds. The decoration A_3 is an injective function, due to uniqueness of labels $\ell(v)$ and due to the fact that, if nodes $u \neq u'$ used the same node $v \in X$ to compute $\lambda(u, v)$ and $\lambda(u', v)$, respectively, then $\lambda(u, v) \neq \lambda(u', v)$. In view of Lemma 2.2, all nodes accomplish (labeled) topology reconstruction within additional $D + 1$ rounds after all nodes have computed their value according to decoration A_3. Hence the algorithm terminates within time $D + 4t + 1 \leq D + k$.

Since the size of the set $N_{t-1}(v)$ is at least t, the size of advice is at most $\lceil (\log n)/t \rceil + 1 \in O(1 + (\log n)/k)$, which completes the proof of the theorem. □

We now provide a lower bound on the minimum size of advice sufficient to perform topology recognition. This bound matches the upper bound given by Algorithm TR-2 in the time-interval $[D + 1, \ldots, 3\lfloor D/2 \rfloor]$.

Theorem 4.2 *Let $2 \leq D \leq \alpha n$ and $0 < k \leq D/2$. The size of advice needed to perform topology recognition in time $D + k$ in the class of graphs of size n and diameter D is in $\Omega((\log n)/k)$.*

Proof: Our lower bound will be proved using the following classes $\mathcal{B}(n, D, k)$ of graphs of size n and diameter D, called *brooms*. We define these classes for n sufficiently large and for $k < \log n$. (For $k \geq \log n$ Proposition 3.2 applies.) Nodes in a broom $B \in \mathcal{B}(n, D, k)$ are partitioned into three sets, called the

bristles, the *stick*, and the *handle*. Let m be the largest even integer such that $km + D - k < n$.

The set bristles consists of km nodes, partitioned into m pairwise disjoint sets $B_1, \ldots, B_m$, each of size k. The stick consists of $D - k$ nodes, and the remaining $n - (km + D - k)$ nodes are in the handle. Hence the bristles, the stick, and the handle are non-empty sets. We now describe the set of edges in each part.

Edges of the bristles are partitioned into two sets, E_1 and E_2. Edges in E_1 connect nodes of each set B_i into a path with port numbers 0 and 1 at each edge. We call *head* of each set B_i the endpoint of the path to which port number 1 has been assigned, and *tail* the other endpoint (to which port number 0 has been assigned). Notice that sets B_i can be of size 1, in which case heads coincide with tails. Edges in E_2 form a perfect matching M among tails of the bristles. These edges have port number 1 at both endpoints.

Edges of the stick form a path of length $D - k - 1$ with port numbers 0 and 1 at each edge. (Notice that this path is of length 0, i.e., the stick consists of a single node, when $D = 2$.) The handle has no edges.

The bristles, the stick, and the handle are connected as follows. Let u be the endpoint of the stick to which port number 1 has been assigned, and let v be the other endpoint of the stick (to which port number 0 has been assigned). Nodes v and u coincide when $D = 2$. Node v is connected to the head of each set B_i by an edge with port numbers i at v and 0 at each head. Node u is connected to each node in the handle. Port numbers at u corresponding to these connecting edges are numbered $\{0, 2, \ldots, n - (km + D - k)\}$, if $u \neq v$, and $\{0, m+1, \ldots, n - (k-1)m - 2\}$, if $u = v$. Nodes in the handle are of degree 1, so they have a unique port with number 0. See Figure 2 for an example of a broom in $\mathcal{B}(23, 6, 3)$. Notice that all brooms in $\mathcal{B}(n, D, k)$ are defined over the same set of nodes and share the same edges, apart from those in sets forming perfect matchings among tails of the bristles. Moreover notice that growing the length of the bristles above $\lfloor D/2 \rfloor$ would result in a graph of diameter larger than D (due to the distance between any pair of unmatched tails of the bristles), which explains the assumption $k \leq D/2$.

For two brooms B' and B'' in $\mathcal{B}(n, D, k)$ we define *corresponding* nodes as follows. Let h' and h'' in B' and B'', respectively, be the nodes in the handles whose only incident edge has port number 0 at both endpoints. Node $u' \in B'$ corresponds to node $u'' \in B''$, if and only if, the (unique) shortest path (defined as the sequence of port numbers on consecutive edges) from h' to u' is the same as the shortest path from h'' to u''.

The idea of the proof is to show that if the size of advice is smaller than $c(\log n)/k$, for a sufficiently small constant $0 < c < 1$, then there exist two brooms in $\mathcal{B}(n, D, k)$, whose corresponding nodes receive the same advice, for which the decorated view at depth $D + k$ of each node in the handle remains the same in both brooms. Since different brooms are non-isomorphic, this will imply the theorem, in view of Proposition 2.1.

Observe that for $k \in \Omega(\log n)$, we have that $c(\log n)/k$ is constant, and $\Omega(1)$ is a lower bound on the size of advice for topology recognition, regardless of the
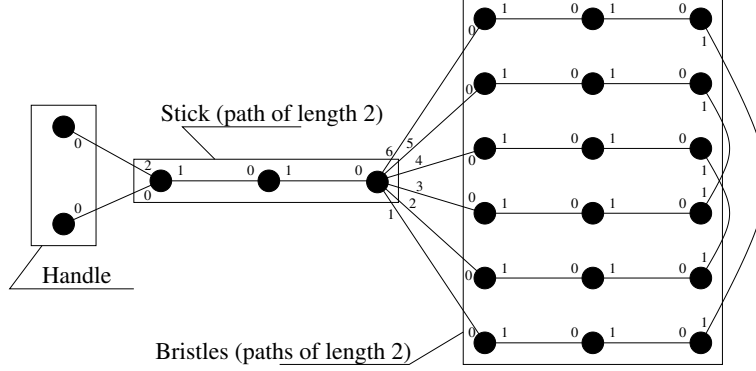
Figure 2: A broom in $\mathcal{B}(23, 6, 3)$.

allowed time. Hence we do not need to define brooms when $k \geq \log n$ to prove the theorem.

We now provide a lower bound on the size of the class $\mathcal{B}(n, D, k)$. This size depends on the number m of tails of the bristles among which perfect matchings M can be defined. For given n and k, the size of the class $\mathcal{B}(n, D, k)$ cannot increase when D grows. Hence the class is smallest for the largest considered value of D, i.e., $D = \lfloor \alpha n \rfloor$. We do the estimation for this value of D.

The number of perfect matchings among tails is at least $(m-1) \cdot (m-3) \cdot (m-5) \cdot \ldots \cdot 3 \cdot 1 > (m/2)!$.

Suppose, from now on, that the size of advice is bounded by $c(\log n)/k$, for some constant $0 < c < 1$. Then there are at most $2^{(c(\log n)/k+1)n}$ ways of assigning advice to nodes of a broom in $\mathcal{B}(n, D, k)$. Hence there are at least $(m/2)!/2^{(c(\log n)/k+1)n}$ brooms in $\mathcal{B}(n, D, k)$ for which corresponding nodes get the same advice. Fix one such assignment A of advice.

We now provide an upper bound on the number of distinct decorated views, at depth $D + k$, of any node in the handle, when advice is assigned to nodes according to A. Consider two brooms B' and B'' in $\mathcal{B}(n, D, k)$, decorated according to assignment A. Let M' and M'' be the perfect matchings among tails of the bristles corresponding to brooms B' and B'', respectively. B' and B'' result in distinct decorated views, at depth $D + k$, of corresponding nodes in the handle, if and only if, there exist corresponding tails of the bristles $t_i' \in B'$ and $t_i'' \in B''$, such that the decorated path B_j', whose tail t_i' is matched to t_i' in M' and the decorated path B_h'', whose tail t_i'' is matched to t_i'' in M'', are different. The number of distinct decorated paths B_i of length $k-1$ is at most $x = 2^{(c(\log n)/k+1)k}$. Since $m \leq n/k$, it follows that there are at most $x^{n/k} = 2^{(c(\log n)/k+1)n}$ distinct decorated views, at depth $D + k$, for any node in the handle, for assignment A.

We will show that the following inequality $(m/2)! > 2^{2(c(\log n)/k+1)n}$ which we denote by (*), holds for $c < (1-\alpha)/128$, when n is sufficiently large. Indeed, for sufficiently large n we have $m > n(1-\alpha)/(2k)$; in view of $k < \log n$, taking

14

the logarithms of both sides we have

$$\log\left(\frac{m}{2}!\right) > \frac{m}{4}\log\frac{m}{4} > \frac{n(1-\alpha)}{8k}\log\frac{n(1-\alpha)}{8k} \geq 2\left(\frac{c\log n}{k}+1\right)n.$$

Inequality (*) implies that $(m/2)!/2^{(c(\log n)/k+1)n} > 2^{(c(\log n)/k+1)n}$. Hence the number of brooms from $\mathcal{B}(n, D, k)$, decorated according to assignment A, exceeds the number of distinct decorated views, at depth $D + k$, of any node in the handle, for these brooms. It follows from the pigeonhole principle that some decorated view corresponds to different brooms from $\mathcal{B}(n, D, k)$. In view of Proposition 2.1, this proves that (even anonymous) topology recognition in time $D + k$, for the class of graphs of diameter D and size n, requires advice of size at least $(1 - \alpha)(\log n)/(128k) \in \Omega(\log n/k)$. $\qquad\square$

Since the lower bound $\Omega(1)$ on the size of advice holds regardless of time, theorems 4.1 and 4.2 imply the following corollary.

Corollary 4.1 *Let $D \leq \alpha n$ and $0 < k \leq D/2$. The minimum size of advice sufficient to perform topology recognition in time $D + k$ in the class of graphs of size n and diameter D is in $\Theta(1 + (\log n/k))$.*

5 Time D

In this section we provide asymptotically tight upper and lower bounds on the minimum size of advice sufficient to perform topology recognition in time equal to the diameter D of the network. Together with the upper bound proved in Theorem 4.1, applied to time $D + 1$, these bounds show an exponential gap in the minimum size of advice due to time difference of only one round.

Algorithm TR-3
Advice:
The oracle assigns a unique label $\ell(u)$ from the set $\{0, \ldots, n-1\}$ to each node u. The advice given to each node u consists of the diameter D, the label $\ell(u)$, and the collection of all edges incident to u, coded as quadruples $\langle \ell(u), p, q, \ell(v) \rangle$, where p is the port number at node u corresponding to edge $\{u, v\}$, and q is the port number at node v corresponding to this edge.
Node protocol:
In round i, each node sends to all its neighbors the collection of edges learned in all previous rounds. After D rounds of communication each node reconstructs the topology and stops. $\qquad\diamond$

Proposition 5.1 *Algorithm TR-3 completes topology recognition for all graphs of size n and diameter D in time D, using advice of size $O(n \log n)$.*

Proof: Since each node is within distance D from all others, each node learns the whole collection of edges of the graph within D rounds. Since these edges are coded using unique labels of adjacent nodes and contain port numbers, this is enough to reconstruct the (labeled) topology.
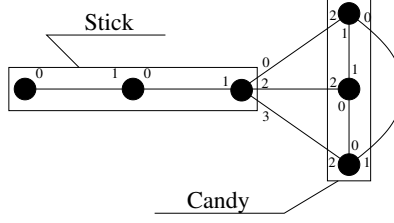
Figure 3: A lollipop in $\mathcal{L}(6,3)$.

Since, in a graph of size n, the diameter can be coded with $O(\log n)$ bits and there are $O(n)$ edges incident to a node, each edge being coded with $O(\log n)$ bits, the size of advice is $O(n \log n)$. □

The following lemma will be used for our lower bound.

Lemma 5.1 *There are at least $((n-1)!)^n/(n!)$ non-isomorphic cliques of size n.*

Proof: Assume first that the nodes are labeled $0, \ldots, n-1$. For each node $i \in [0, n-1]$ there are $(n-1)!$ ways to assign ports at it. These assignments are independent for each node, hence there are $((n-1)!)^n$ distinct assignments of port numbers. Since from one clique without node labels it is possible to obtain at most $n!$ distinct cliques with nodes labeled $0, \ldots, n-1$, our lower bound follows. □

We define the following classes $\mathcal{L}(n, D)$ of graphs of size n and diameter $D \leq \alpha n$, called *lollipops*. These graphs will be used to prove our lower bounds for time D and below. Nodes in a lollipop $L \in \mathcal{L}(n, D)$ are partitioned into two sets, called the *candy* and the *stick*. The candy consists of $n - D$ nodes; for the purpose of describing our construction we will call these nodes $w_1, \ldots, w_{n-D}$. The stick consists of the remaining D nodes (see fig. 3 for an example of a lollipop in $\mathcal{L}(6,3)$).

Nodes in the candy are connected to form a clique; port numbers for these edges are assigned arbitrarily from the set $\{0, \ldots, n - D - 2\}$. Edges of the stick form a path of length $D - 1$ with port numbers 0 and 1 at each edge.

The stick and the candy are connected as follows. Let v be the endpoint of the stick to which port number 1 has been assigned and let u be the other endpoint of the stick (to which port number 0 has been assigned). Notice that u and v coincide, when $D = 1$. Node v is connected to all nodes in the candy. The port number, at node v, corresponding to edge $\{v, w_i\}$ is 0, if $i = 1$. For $i > 1$ this port number is i, when $u \neq v$ and $i - 1$, when $u = v$. The port number, at all nodes w_i, corresponding to edge $\{v, w_i\}$ is $n - D - 1$.

Since, for $D \leq \alpha n$, the size of the candy of a lollipop in $\mathcal{L}(n, D)$ is at least $\lceil n(1 - \alpha) \rceil$, Lemma 5.1 implies the following corollary:

Corollary 5.1 *The size of the class $\mathcal{L}(n, D)$, for $D \leq \alpha n$ is at least $((\lceil n(1 - \alpha) \rceil - 1)!)^{\lceil n(1-\alpha) \rceil}/(\lceil n(1 - \alpha) \rceil!)$.*

Theorem 5.1 *Let $D \leq \alpha n$. The size of advice needed to perform topology recognition in time D in the class of graphs of size n and diameter D is in $\Omega(n \log n)$.*

Proof: If we consider a lollipop of diameter $D \leq \alpha n$, then there are $\Omega(n^2)$ edges of the candy that are outside of the view at depth D of the endpoint u of the stick. The idea of the proof is based on the fact that information about these edges has to be assigned to nodes of the graph as advice that will become available to u within time D.

First observe that the view at depth D of the endpoint u of the stick is the same for all lollipops in $\mathcal{L}(n, D)$. Hence, if the size of advice is at most $cn \log n$, then the number of distinct decorated views of node u is at most $2^{cn^2 \log n + n} < 2^{cn^2(\log n + 1)}$. We will show that, if $c < (1 - \alpha)^2/5$, then, for sufficiently large n, the number of lollipops in $\mathcal{L}(n, D)$ exceeds this bound. Indeed, by Corollary 5.1, the size of the class $\mathcal{L}(n, D)$ is at least

$$\frac{(\lceil n(1 - \alpha) \rceil - 1)!^{\lceil n(1-\alpha) \rceil}}{\lceil n(1 - \alpha) \rceil!}$$

For sufficiently large n we have

$$\frac{(\lceil n(1 - \alpha) \rceil - 1)!^{\lceil n(1-\alpha) \rceil}}{\lceil n(1 - \alpha) \rceil!} > \frac{n(1 - \alpha)}{2}^{n^2(1-\alpha)^2/4}.$$

It is enough to show that

$$\left(\frac{n(1 - \alpha)}{2} \right)^{n^2(1-\alpha)^2/4} > 2^{n^2(\log n + 1)(1-\alpha)^2/5},$$

which is immediate to verify by taking the logarithm of both sides.

It follows that the same decorated view at depth D of node u corresponds to different lollipops from $\mathcal{L}(n, D)$. In view of Proposition 2.1, this proves that (even anonymous) topology recognition in time D, for the class of graphs of size n and diameter $D \leq \alpha n$, requires advice of size at least $(n \log n)(1 - \alpha)^2/5 \in \Omega(n \log n)$. $\qquad\square$

Proposition 5.1 and Theorem 5.1 imply the following corollary.

Corollary 5.2 *Let $D \leq \alpha n$. The minimum size of advice sufficient to perform topology recognition in time D in the class of graphs of size n and diameter D is in $\Theta(n \log n)$.*

6 Time below D

In this section we study the minimum size of advice sufficient to perform topology recognition when the time allotted for this task is too short, for some node, to communicate with all other nodes in the network.

17

Algorithm TR-4

Advice:

Let G be a graph of size n and diameter D. The oracle assigns a unique label $\ell(v)$ from $\{0, \ldots, n-1\}$ to each node v in the graph G. It codes all edges of the graph as quadruples $\langle \ell(u), p, q, \ell(v) \rangle$, where $\ell(u)$ and $\ell(v)$ are the labels of two adjacent nodes u and v, p is the port number at node u corresponding to edge $\{u, v\}$, and q is the port number at node v corresponding to this edge. Let E be the set of all these codes.

Let $t = \lfloor (D - k)/3 \rfloor$, for $D \geq k > 0$, be the time available to complete the topology recognition. If $t = 0$, then the advice provided by the oracle to each node u is: $\ell(u)$, the collection E of all edges, and the integer 0.

If $t \geq 1$, then the oracle picks a set X of nodes in G satisfying Lemma 2.1, for $r = 2t$. For each node $x \in X$, let $z(x) = |N_t(x)|$. Moreover, let $E_1, \ldots, E_{z(x)}$ be a partition of the edges in E into $z(x)$ pairwise disjoint sets of sizes differing by at most 1. Let $v_1, \ldots, v_{z(x)}$ be an enumeration of nodes in $N_t(x)$. The advice given by the oracle to node $v_i \in N_t(x)$ consists of the label $\ell(v_i)$, of the set E_i, and of the integer t. Every other node u only gets $\ell(u)$ and t as advice. Let A be the resulting assignment of advice.

Node protocol:

Let t be the integer received by all nodes as part of their advice.

In round i, with $1 \leq i \leq 3t$, each node sends to all its neighbors the collection of edges learned in all previous rounds. (In particular, if $t = 0$, then there is no communication.) After $3t$ rounds of communication each node reconstructs the topology and stops. $\diamond$

Theorem 6.1 *Let $0 < k \leq D$. Algorithm* TR-4 *completes topology recognition for all graphs of size n and diameter D within time $D - k$, using advice of size $O((n^2 \log n)/(D - k + 1))$.*

Proof: If $t = 0$ then $D - k$ is constant. In this case each node uses the collection E of edges received as advice to reconstruct the topology in time 0. Since there are $O(n^2)$ edges in a graph of size n, and each edge can be coded with $O(\log n)$ bits, the size of advice is $O(n^2 \log n) = O((n^2 \log n)/(D - k + 1))$.

Hence we can assume that $t \geq 1$. Since each node is at distance at most $2t$ from some node in X, and the decorations of the nodes in $N_t(v)$, for each such node v, assigned according to A, collectively contain the entire set E, after time $3t \leq D - k$ each node of the graph can reconstruct the whole (labeled) topology. It remains to be shown that the size of advice is in $O((n^2 \log n)/(D - k + 1))$. Indeed, $z(x) > t \in \Omega(D - k + 1)$ and the size of each set E_i is at most $\lceil |E|/z(x) \rceil$. Each edge can be coded with $O(\log n)$ bits and sets $N_t(x)$, for $x \in X$, are disjoint. The theorem follows from the fact that $|E| \in O(n^2)$. $\square$

The following lower bound shows that the size of advice used by Algorithm TR-4 is asymptotically optimal.

Theorem 6.2 *Let $D \leq \alpha n$ and $0 < k \leq D$. The size of advice needed to perform topology recognition in time $D - k$ in the class of graphs of size n and diameter D is in $\Omega((n^2 \log n)/(D - k + 1))$.*

Proof: The argument follows closely the proof of Theorem 5.1. The only difference is that we consider decorated views of the endpoint u of the stick at depth $D - k$ instead of D. Since there are only $D - k + 1$ nodes in the lollipop at distance $D - k$ from u, if the size of advice is at most $(cn^2 \log n)/(D - k + 1)$, then the number of distinct decorated views of node u at depth $D - k$ is at most $2^{((cn^2 \log n)/(D-k+1)+1)(D-k+1)} = 2^{cn^2 \log n + (D-k+1)} \le 2^{cn^2(\log n + 1)}$, for sufficiently large n.

The same computation as in the proof of Theorem 5.1 shows that, if $c < (1-\alpha)^2/7$, then the same decorated view at depth $D - k$ of node u corresponds to different lollipops from $\mathcal{L}(n, D)$. In view of Proposition 2.1, this proves that (even anonymous) topology recognition in time $D - k$, for the class of graphs of diameter D and size n, requires advice of size at least $((1-\alpha)^2/7) \cdot (n^2 \log n)/(D - k + 1) \in \Omega((n^2 \log n)/(D - k + 1))$. $\square$

Theorems 6.1 and 6.2 imply the following corollary.

Corollary 6.1 *Let $D \le \alpha n$ and $0 < k \le D$. The minimum size of advice sufficient to perform topology recognition in time $D - k$ in the class of graphs of size n and diameter D is in $\Theta((n^2 \log n)/(D - k + 1))$.*

7   Conclusion and open problems

We presented upper and lower bounds on the minimum size of advice sufficient to perform topology recognition, in a given time T, in n-node networks of diameter D. Our bounds are asymptotically tight for time $T = 2D + 1$ and, if $D \le \alpha n$ for some constant $\alpha < 1$, in the time interval $[0, \ldots, 3D/2]$. Moreover, in the remaining time interval $(3D/2, \ldots, 2D]$ our bounds are still asymptotically tight if $D \in \Omega(\log n)$. Closing the remaining gap between the lower bound 1 and the upper bound $O(1 + (\log n)/k)$ in this remaining time interval, for graphs of very small diameter $D \in o(\log n)$, is a natural open problem. In particular, it would be interesting to find the minimum time in which topology recognition can be accomplished using advice of constant size, or even of size exactly 1.

Other open problems remain in the case of networks with very large diameter, those which do not satisfy the assumption $D \le \alpha n$ for some constant $\alpha < 1$, or equivalently those for which $n - D \in o(n)$. Our upper bounds do not change in this case (we did not use the assumption $D \le \alpha n$ in their analysis), while our lower bounds change as follows, using the same constructions. The lower bound for time above D, i.e., when $T = D + k$, where $0 < k \le D$, becomes $\Omega((\log(n - D))/k)$; our lower bound for time D becomes $\Omega(((n - D)^2 \log(n - D))/n)$; the lower bound for time below D, i.e., when $T = D - k$, where $0 < k \le D$, becomes $\Omega(((n - D)^2 \log(n - D))/(D - k + 1))$. It remains to close the gaps between these lower bounds and the upper bounds that we gave for each allotted time.

Another open problem is related to the communication model used. As mentioned in the Introduction, we chose the $\mathcal{LOCAL}$ model which ignores the size of messages. It would be interesting to study how the results change in the $\mathcal{CONGEST}$ model in which messages are limited to logarithmic size. Some

compression of the information transmitted should be possible, e.g., along the lines of [29].

Let us also address the issue of node identities vs. advice given to nodes. We did our study for unlabeled networks, arguing that nodes may be reluctant to disclose their identities for security or privacy reasons. As we have seen, however, for anonymous networks some advice has to be given to nodes, regardless of the allotted time. Does the oracle have to provide new distinct labels to nodes? Our results show that for time above D this is not the case, as the minimum size of advice enabling topology recognition in this time is too small for assigning a unique identifier to each node. Hence, in spite of not having been given, a priori, unique identifiers, nodes can perform labeled topology recognition in this time span. On the other hand for time at most D, the minimum size of advice is sufficiently large to provide distinct identifiers to nodes, and indeed our oracles inserted unique identifiers as part of advice. However, this should not raise concerns about security or privacy, as these identifiers may be arbitrary and hence should be considered as "nicknames" temporarily assigned to nodes.

A different set of problems would appear if (unlike in our scenario) randomization were allowed. For many time values the size of required advice would then be drastically reduced. For example, for time D, advice of size $O(\log n)$ would suffice. Indeed, providing nodes with the diameter D and the size n of the network would enable them to first acquire different labels with high probability (the size n, or some bound on it, is needed to know how long the random labels should be, but such labels can be randomly produced with no communication), and then learn the topology of the resulting labeled graph in D rounds. We leave the analysis of tradeoffs between time and advice for randomized topology recognition as an open problem.

References

[1] S. Abiteboul, H. Kaplan, T. Milo, Compact labeling schemes for ancestor queries, Proc. 12th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2001), 547–556.

[2] D. Angluin, Local and global properties in networks of processors. Proc. 12th Annual ACM Symposium on Theory of Computing (STOC 1980), 82–93.

[3] H. Attiya, M. Snir, M. Warmuth, Computing on an anonymous ring, Journal of the ACM 35, (1988), 845–875.

[4] H. Attiya, A. Bar-Noy, D. Dolev, D. Koller, D. Peleg, R. Reischuk, Renaming in an asynchronous environment, Journal of the ACM 37 (1990), 524–548.

[5] B. Awerbuch, Optimal distributed algorithms for minimum weight spanning tree, counting, leader election and related problems, Proc. 19th Annual ACM Symposium on Theory of Computing (STOC 1987), 230–240.

[6] P. Boldi, S. Vigna, Computing anonymously with arbitrary knowledge, Proc. 18th ACM Symposium on Principles of Distributed Computing (PODC 1999), 181–188.

[7] S. Caminiti, I. Finocchi, R. Petreschi, Engineering tree labeling schemes: a case study on least common ancestor, Proc. 16th Annual European Symposium on Algorithms (ESA 2008), 234–245.

[8] J. Chalopin, S. Das, A. Kosowski, Constructing a map of an anonymous graph: Applications of universal sequences, Proc. 14th International Conference on Principles of Distributed Systems (OPODIS 2010), 119–134.

[9] R. Cohen, P. Fraigniaud, D. Ilcinkas, A. Korman, D. Peleg, Label-guided graph exploration by a finite automaton, ACM Transactions on Algorithms 4 (2008).

[10] D. Dereniowski, A. Pelc, Drawing maps with advice, Journal of Parallel and Distributed Computing 72 (2012), 132–143.

[11] Y. Emek, P. Fraigniaud, A. Korman, A. Rosen, Online computation with advice, Theoretical Computer Science 412 (2011), 2642–2656.

[12] Y. Emek, C. Pfister, J. Seidel, R. Wattenhofer, Anonymous networks: Randomization= 2-hop coloring, Proc. 32nd ACM Symposium on Principles of Distributed Computing (PODC 2014), 96–105.

[13] Y. Emek, J. Seidel, R. Wattenhofer, Computability in anonymous networks: Revocable vs. irrecovable outputs, Proc. 41th International Colloquium on Automata, Languages and Programming (ICALP 2014), 183-195.

[14] P. Fraigniaud, C. Gavoille, D. Ilcinkas, A. Pelc, Distributed computing with advice: Information sensitivity of graph coloring, Distributed Computing 21 (2009), 395–403.

[15] P. Fraigniaud, D. Ilcinkas, A. Pelc, Communication algorithms with advice, Journal of Computer and System Sciences 76 (2010), 222–232.

[16] P. Fraigniaud, D. Ilcinkas, A. Pelc, Tree exploration with advice, Information and Computation 206 (2008), 1276–1287.

[17] P. Fraigniaud, A. Korman, E. Lebhar, Local MST computation with short advice, Theory of Computing Systems 47 (2010), 920–933.

[18] E. Fusco, A. Pelc, Trade-offs between the size of advice and broadcasting time in trees, Algorithmica 60 (2011), 719–734.

[19] E. Fusco, A. Pelc, How much memory is needed for leader election, Distributed Computing 24 (2011), 65–78.

[20] C. Gavoille, D. Peleg, S. Pérennes, R. Raz. Distance labeling in graphs, Journal of Algorithms 53 (2004), 85-112.

[21] D.S. Hirschberg, J.B. Sinclair, Decentralized extrema-finding in circular configurations of processes, Communications of the ACM 23 (1980), 627–628.

[22] D. Ilcinkas, D. Kowalski, A. Pelc, Fast radio broadcasting with advice, Theoretical Computer Science, 411 (2012), 1544–1557.

[23] M. Katz, N. Katz, A. Korman, D. Peleg, Labeling schemes for flow and connectivity, SIAM Journal of Computing 34 (2004), 23–40.

[24] A. Korman, S. Kutten, D. Peleg, Proof labeling schemes, Distributed Computing 22 (2010), 215–233.

[25] E. Kranakis, D. Krizanc, J. van der Berg, Computing Boolean functions on anonymous networks, Information and Computation 114 (1994), 214–236.

[26] N. Nisse, D. Soguet, Graph searching with advice, Theoretical Computer Science 410 (2009), 1307–1318.

[27] D. Peleg, Distributed computing, a locality-sensitive approach, SIAM Monographs on Discrete Mathematics and Applications, Philadelphia 2000.

[28] G.L. Peterson, An $O(n \log n)$ unidirectional distributed algorithm for the circular extrema problem, ACM Transactions on Programming Languages and Systems 4 (1982), 758–762.

[29] S. Tani, Compression of view on anonymous networks - Folded view -. IEEE Trans. Parallel Distrib. Syst. 23 (2012), 255-262.

[30] M. Thorup, U. Zwick, Approximate distance oracles, Journal of the ACM, 52 (2005), 1–24.

[31] M. Yamashita, T. Kameda, Computing on anonymous networks: Part I - characterizing the solvable cases, IEEE Transactions on Parallel and Distributed Systems 7 (1996), 69–89.