Views in a graph: to which depth must equality be checked?

Julien M. Hendrickx

***Abstract*—The *view of depth* k of a node is a tree containing all the walks of length k leaving that node. Views contain all the information that nodes could obtain by exchanging messages with their neighbors. In particular, a value can be computed by a node on a network using a distributed deterministic algorithm if and only if that value only depends on the node's view of the network.**

Norris has proved that if two nodes have the same view of depth $n-1$, they have the same views for all depths. Taking the diameter d into account, we prove a new bound in $O(d + d\log(n/d))$ instead of $n-1$ for bidirectional graphs with port numbering, which are natural models in distributed computation. This automatically improves various results relying on Norris's bound. We also provide a bound that is stronger for certain colored graphs and extend our results to graphs containing directed edges.

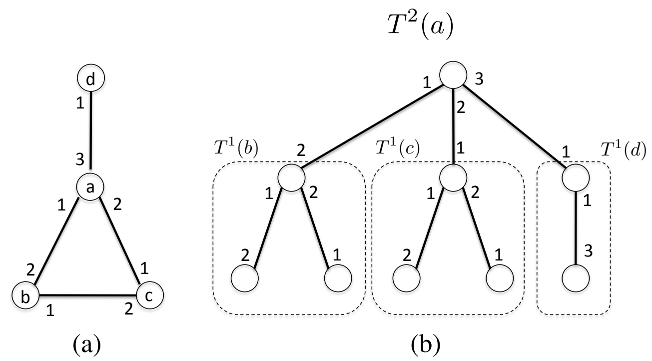


Figure 1. Example of a bidirectional graph with port-numbering (a), and the view of depth 2 of the node a in that graph (b). The view of depth 2 of a is obtained by connecting a root node to the root node of the view of depth 1 of the neighbors of a by edges having the same port numbers (in both directions) as in the initial graph. Nodes b and c have the same view of depth 1 ($T^1(b) = T^1(c)$), but one can verify that their views of depth 2 are different. Note that the labels a, b, c, d are used to distinguish the nodes when describing the graph, but are unknown to the nodes, and are not part of their views.

I. Introduction

A graph with port numbering is a graph where nodes have locally unique numbers assigned to their incident edges, allowing them to distinguish their neighbors, as in the example shown in Figure 1(a). For such graphs, the view of a node is an infinite rooted tree that represents all the infinite walks starting at that node in the graph together with the port numbers encountered on these paths (see Figure 1(b)), and that is locally isomorphic to the initial graph. Views have been introduced by Yamashita and Kameda [16], who proved that they contain all the information about the graph that the node could obtain by exchanging messages with its neighbors (see for example Lemma 5 in [16] or Theorem 5 in [12]). In particular, if two nodes have the same view, they are undistinguishable from the point of view of distributed algorithms, and the execution of any distributed deterministic algorithm will always leave them in identical states. As a result, a value can be computed by a node on a network using a distributed deterministic algorithm if and only if that value only depends on the node's view of the network.

Views are infinite objects, but Yamashita and Kameda have shown that if the truncation at depth n^2 of the views of two nodes on a network of n nodes are equal, then their whole views are equal. This bound was later improved by Norris [13], who showed that equality of the views truncated at depth $n-1$ (or "views of depth $n-1$") was sufficient to guarantee equality of the views. Her result was actually proved for the more general context of universal cover of directed graphs with arbitrary edge labels.

This bound plays a fundamental role in the development, the validation, and the analysis of many distributed algorithms, see e.g. [1], [3]–[11], [14]–[17]. In particular, it can be used to prove that the view of depth $2n-1$ of a node, actually contains all the information that could be made available to that node. (The same holds true for depth $n+d$, where d is the diameter of the graph). This can be used to bound the number of messages that need to be passed in certain distributed algorithms and the communication and computation cost at each node, see Section IV-A for more details.

Norris's bound is tight, in the sense that there exist families of graphs where some nodes have equal views for all depths smaller than $n-1$, but different views of depth $n-1$ [2], [13]. Stronger bounds involving other quantities may however exist. Fraignaud and Pelc [11] have for example proved the bound $\hat{n}-1$ where $\hat{n}$ is the number of different views present in the network, with $\hat{n}-1 = n-1$ when all nodes have different views.

In this work, we improve Norris's bound for bidirectional graphs with port numbering by taking the diameter into account. We prove a family of bounds $t-1+d+(d+1)\lfloor \log_2 \frac{n}{t} \rfloor$ for every integer $t \leq n$, and derive from this family the slightly weaker bound $(d+1)\left(1.914 + \log_2\left(\frac{n}{d+1}\right)\right) - 1$ (for $d \leq n \ln 2 - 1 \simeq 0.69n - 1$). We also prove a bound that is stronger for certain colored graphs, and extend our results to some classes of directed graphs. Our results do not contradict

Julien M. Hendrickx is with the ICTEAM Institute at the Université catholique de Louvain, Avenue Georges Lemaitre 4, B-1348 Louvain-la-Neuve, Belgium; julien.hendrickx@uclouvain.be. This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author. The author wishes to thank John Tsitsiklis for discussions having led to the discovery of the problem.

the tightness of Norris's bound, as the graphs on which it is tight have diameters $d = n - 1$. For graphs with diameters smaller than n however, our result can lead to much smaller bounds, which automatically improves various results that rely on Norris's bound.

II. Problem Definition

Similarly to [16], we consider a bidirectional graph $G(V, E)$ on $n = |V|$ nodes, where nodes may be colored in an arbitrary way. In addition, each edge (v, w) has a port number: for each node v there is an injective function σ_v defined on its set of incident edges, as in Figure 1(a). In most works, σ_v is actually a bijection taking its values in $\{1, \ldots, deg_v\}$, but this is not relevant in our context. We call $\sigma_v(v, w)$ the *port number* of the edge (v, w). For the sake of concision, we will not explicitly mention node coloring and port numbering when referring to a graph G, but it should be understood that every graph considered here comes with a node-coloring and a port-numbering.

From a distributed computation point of view, assuming the presence of a port numbering corresponds to assuming that every node has a local way of identifying its neighbors, and of knowing by which number each of its neighbors identifies it thanks to the bidirectionality of the edges. The node colors represent all the additional information available to the nodes, including variables initially stored in memory, partial identifiers and other intrinsic properties of the nodes. A graph without colors represents thus a system where nodes initially have no additional information. Node colors may help distinguishing nodes, and are even sometimes essential to break symmetries.

The notion of view is defined recursively. The *view of depth* 0 $T^0(v)$ of a node v, consists of a node called *the root*, having the same color as v in G. *The view of depth* k of v, $T^k(v)$, is obtained by taking a root node with the same color as v, and for every neighbor v_i of v, (i) the view of depth $k-1$ of v_i, and (ii) an edge connecting the root node r to the root $r(T^{k-1}(v_i))$ of $T^{k-1}(v_i)$ with the same port-numbers as the edge (v, v_i), i.e. $\sigma_r(r, r(T^{k-1}(v))) = \sigma_v(v, v_i)$ and $\sigma_{r(T^{k-1}(v))}(r(T^{k-1}(v)), r) = \sigma_{v_i}(v_i, v)$, as represented in Figure 1(b). (Port-numbers can be formally defined on views as the combination for each node of (i) an injective function defined on its set of edges going away from the root, and (ii) one "incoming port number" for the edge connecting it to its parent node, closer from the root, except if there is no such edge because the node is the root of the view.)

One can easily see by recurrence that $T^k(v)$ is a subgraph of $T^{k+1}(v)$ and that they share the same root. We can then define the *(infinite) view* $T(v)$ as the infinite rooted tree with port numbering resulting from the countable union $\bigcup_{k\geq 0} T^k(v)$.

We define $B_{n,d}$ as the smallest m for which $T^m(v) = T^m(w) \Leftrightarrow T(v) = T(w)$ holds for every two nodes v, w of every graph G on n nodes with diameter d (where $T(v) = T(w)$ is equivalent to $T^k(v) = T^k(w)$ for every k). Norris's bound implies that $B_{n,d}$ is well defined and that $B_{n,d} \leq n - 1$. We will provide a new bound.

III. Results

Our bound relies on two intermediate results. The first one was established in [13] for a larger class of graphs. We present a short proof here for the sake of completeness and because it helps developing intuition about certain aspects of our result. We define the equivalence relation $\sim_k$ between nodes by saying that $v \sim_k w$ if their views of depth k are equal: $T^k(v) = T^k(w)$. We then define π_k as the partition induced by $\sim_k$ on the set of nodes, and call "blocks" the classes that this partition defines, that is, the equivalence classes induced by $\sim_k$.

Lemma 1 (Norris [13]).
(a) π_{k+1} is a refinement of π_k: if v and w are in distinct blocks of π_k then they are in distinct blocks of π_{k+1}.
(b) If $\pi_{k+1} = \pi_k$ for some $k \geq 0$ then $\pi_j = \pi_k$ for all $j > k$.

Proof. Part (a) directly follows from the fact that two nodes having different views of depth k obviously have different views of depth $k+1$ since the latter contain the former. To prove part (b), we just need to prove that $\pi_{k-1} = \pi_k$ implies $\pi_k = \pi_{k+1}$ for any $k > 0$ and the rest will follow by recurrence. Let us thus suppose that $\pi_{k-1} = \pi_k$, i.e. $T^{k-1}(v') = T^{k-1}(v'') \Leftrightarrow T^k(v') = T^k(v'')$, and consider two arbitrary nodes v, w. By definition of the view, $T^k(v) = T^k(w)$ holds for $k > 0$ if and only if the three following conditions are satisfied:

1) v and w have the same color (if any) and the same degree.
2) There is a one-to-one correspondence between the neighbors v_i of v and w_i of w such that if v_i corresponds to w_i, then (v, v_i) and (w, w_i) have the same port numbers: $\sigma_v(v, v_i) = \sigma_w(w, w_i)$ *and* $\sigma_{v_i}(v_i, v) = \sigma_{w_i}(w_i, w)$.
3) For every pair v_i, w_i defined above, there holds $T^{k-1}(v_i) = T^{k-1}(w_i)$.

The first two conditions are independent of k as long as $k > 0$, and thus remain satisfied for depth $k+1$ if they are satisfied at depth k. The last condition does depend on k. But, under the assumption that $\pi_k = \pi_{k-1}$, we know that $T^{k-1}(v_i) = T^{k-1}(w_i)$ if and only if $T^k(v_i) = T^k(w_i)$, so that the third condition is satisfied at depth $k+1$ if and only if it was satisfied at depth k. As a consequence $T^{k+1}(v) = T^{k+1}(w) \Leftrightarrow T^k(v) = T^k(w)$, and $\pi_k = \pi_{k+1}$. □

We now turn to the second intermediate result, in which we show that the size of any block of of the partition π_k dominates the size of every block of π_{k+d}, where d is the diameter of the graph. Unlike Lemma 1, this result does rely on the local uniqueness of the port numbers.

Let p be a path from v to w, i.e. a sequence of $|p|$ edges $((v, u_1), (u_1, u_2), \ldots, (u_{|p|-1}, w))$. We define the *port sequence of* p as the sequence

$$\lambda_p = \left(\sigma_v(v, u_1), \sigma_{u_1}(u_1, u_2), \ldots, \sigma_{u_{|p|-1}}(u_{|p|-1}, w)\right).$$

Intuitively, λ_p contains the directions to be followed at each node in order to follow the path p. For example, the port sequence of the path $((d,a),(a,c),(c,b))$ in Figure 1 is $(1,2,2)$. The following Lemma, stating that a port sequence together with a starting node uniquely specifies (at most) one path, follows immediately from the injectivity of the port numbers.

Lemma 2. *Two paths p_a and p_b starting at a same vertex are identical if and only if they have the same port sequence $\lambda(p_a) = \lambda(p_b)$.*

The notion of port sequence can easily be extended to paths in the views. In Figure 1 for example, the path going from the lower right-hand side node (corresponding to d) to the root of $T^2(a)$ has a port sequence $(3,1)$. The following Lemma linking port sequences in graphs and in views follows directly from this extension.

Lemma 3. *Let $\tilde{T}^q$ be a view of depth q, $\tilde{\lambda}$ a port sequence of length $\left|\tilde{\lambda}\right|$, and v a node in a graph G. The following two conditions are equivalent.*
a) In the graph G, there is a path with port sequence $\tilde{\lambda}$ starting at some node w with $T^q(w) = \tilde{T}^q$ and arriving at v.
b) In the view $T^{q+|\tilde{\lambda}|}(v)$ there is a path with port sequence $\tilde{\lambda}$ starting from the root of a copy of $\tilde{T}^q$ and arriving at the root of $T^{q+|\tilde{\lambda}|}(v)$.

As an illustration of this Lemma, consider the graph of Figure 1. Take $\tilde{\lambda} = (2)$, and a view $\tilde{T}^1$ where the root is connected to two leaves by edges with port numbers 1 and 2, and "arrival" port numbers 2 and 1 respectively. If a is taken as node v, condition (a) corresponds to the existence of a path with port sequence $\tilde{\lambda} = (2)$ from b to a, with $T^1(b) = \tilde{T}^1$. Condition (b) corresponds to the existence in $T^2(a)$ of a path with port sequence $\tilde{\lambda} = (2)$ from the root of a copy of $\tilde{T}$ to the root of $T^2(a)$. We can now state our second intermediate result.

Lemma 4. *Let G be a connected graph with diameter d. The size of any block of π_k is larger than or equal to the size of all blocks of π_{k+d}.*

Proof. Let B be a block of π_{k+d} and C be a block of π_k. We show that $|B| \leq |C|$ by associating to each node of B a distinct node of C.

Let v and w be arbitrary nodes in B and C respectively, and p a path of length $|p| \leq d$ starting at w and arriving at v, with port sequence $\lambda(p)$. It follows from Lemma 3 that the view $T^{k+d}(v)$ contains a path with port sequence $\lambda(p)$ arriving at its root and starting from the root of a copy of $T^{k+d-|p|}(w)$.

Let now v' be an arbitrary node of B. Its view of depth $k+d$ is by definition the same as that of v, $T^{k+d}(v) = T^{k+d}(v')$, and contains thus also a path with port sequence $\lambda(p)$ arriving at its root and starting from the root of a copy of $T^{k+d-|p|}(w)$. Lemma 3 implies then the existence of a path p' with same port sequence $\lambda(p)$ arriving at v' and starting from some node w' whose view of depth $k+d-|p|$ is the same as that of w, $T^{k+d-|p|}(w') = T^{k+d-|p|}(w)$. This implies that w and w' also have the same view of depth k because $d - |p| \geq 0$, and thus that w' belongs by definition to C.

We can thus associate a node $w' \in C$ to every node $v' \in B$. Consider now two nodes $v_1', v_2' \in B$, their associated nodes $w_1', w_2' \in C$, and the corresponding paths p_1', p_2'. Since $\lambda(p_1') = \lambda(p) = \lambda(p_2')$, it follows from Lemma 2 that if $w_1' = w_2'$, then p_1' and p_2' are identical and have the same arrival node $v_1' = v_2'$. Therefore, $v_1' \neq v_2'$ implies $w_1' \neq w_2'$. We have thus shown that to each node $v' \in B$ is associated a *distinct* node $w' \in C$, which implies that $|B| \leq |C|$. □

Theorem 1. *Let G be a connected bidirectional graph with port numbering on n nodes with diameter d. For every $t = 1, \ldots, n$, two nodes v, w have the same view if and only if they have the same view of depth $t-1+d+(d+1)\lfloor \log_2 \frac{n}{t} \rfloor$. Therefore we have the bound*

$$B_{n,d} \leq t - 1 + d + (d+1)\lfloor \log_2 \frac{n}{t} \rfloor, \qquad (1)$$

where we recall that $B_{n,d}$ is the minimal value m such that, $T^m(v) = T^m(w) \Rightarrow T(v) = T(w)$ for any two nodes v, w of a bidirectional graph with port numbering on n nodes with diameter d.

Proof. Consider such a graph G. If $\pi_{m+1} = \pi_m$ for some $m > 0$, then it follows from Lemma 1(b) that $\pi_k = \pi_m$ for every $k \geq m$. In particular, two nodes have the same (infinite) view if and only if they have the same view of depth m (or the same view of depth k for any arbitrary $k \geq m$), and there holds $B_{n,d} \leq m$.

We now fix now an integer t, and show that there always exists such an m smaller than or equal to $t - 1 + d + (d + 1)\lfloor \log_2 \frac{n}{t} \rfloor$, which will prove the result of this theorem. For this purpose, we show that in case $\pi_{k+1} \neq \pi_k$ holds for every $k < t - 1 + d + (d+1)\lfloor \log_2 \frac{n}{t} \rfloor$ (which we do not claim is actually always possible), then $\pi_{m+1} = \pi_m$ must hold for $m = t-1+d+(d+1)\lfloor \log_2 \frac{n}{t} \rfloor$. The proof of that implication is organized in three claims.

Claim 1: Every block of π_{t-1+d} contains n/t nodes or less.

Lemma 1(a) states that π_{k+1} is a refinement of π_k, i.e. nodes in distinct blocks of π_k are also in distinct blocks of π_{k+1}. The assumption that $\pi_{k+1} \neq \pi_k$ implies then that π_k contains at least one more block than π_k. As a result, π_{t-1} contains t blocks or more since π_0 contains 1 block (or more). At least one of these blocks must thus contains $\frac{n}{t}$ nodes or less, since the total number of nodes is n. Claim 1 follows then directly from Lemma 4.

Claim 2: Every block of $\pi_{t-1+d+s(1+d)}$ has a size $2^{-s}\frac{n}{t}$ or less (for any integer $s \leq \log_2 \frac{n}{t}$).

Since we have assumed that $\pi_{t+d} \neq \pi_{t-1+d}$ (because $t - 1 + d$ is smaller than the expression in (1)), it follows again from Lemma 1(a) that the blocks of π_{t+d} can be obtained by partitioning the blocks of π_{t+d-1}, with at least one partition being nontrivial, that is, at least one block of π_{t-1+d} yielding two or more blocks in π_{t+d}. Consider one of the blocks being the object of a nontrivial partition. Since it contains at most $\frac{n}{t}$ nodes, its partition yields at least one block of $\frac{n}{2t}$ nodes or less. Lemma 4 implies then that $\pi_{t-1+d+1+d}$ only contains blocks of $\frac{n}{2t}$ elements or less. Repeating this argument, we see that all blocks of $\pi_{t-1+d+(1+d)s}$ contain $2^{-s}\frac{n}{t}$ nodes or less, for any $s \leq \lfloor \log_2 \frac{n}{t} \rfloor$ (for larger s, it is not assumed that $\pi_{k+1} \neq \pi_k$).

Claim 3: $\pi_{m+1} = \pi_m$ *for* $m = t - 1 + d + (d+1)\lfloor \log_2 \frac{n}{t} \rfloor$.

By taking $s = \lfloor \log_2 \frac{n}{t} \rfloor$ in Claim 2, we see that the size of every block of the partition π_m is bounded by $\frac{n}{t} 2^{-\lfloor \log_2 \frac{n}{t} \rfloor} < 2$ nodes, and is thus exactly 1 since it is an integer. These blocks can thus not be separated into smaller blocks. Since Lemma 1(a) implies that the blocks of π_{m+1} can be obtained by partitioning those of π_m, there must hold $\pi_{m+1} = \pi_m$, which concludes the proof. □

The bound of Theorem 1 depends on a parameter t, unrelated to the initial problem, and whose value can be set arbitrarily. Our strongest bound on on $B_{n,d}$ is thus obtained by minimizing the bound of Theorem 1 over t, for each couple n, d. In the next corollary, we derive a closed form upper approximation of the solution to that optimization problem.

Corollary 1. *Let* $B_{n,d}$ *be defined as in Theorem 1. If* $d \leq n \ln 2 - 1$*, there holds*

$$B_{n,d} \leq (d+1) \log_2 \left((2e \ln 2) \frac{n}{d+1} \right) - 1 \qquad (2)$$
$$\leq (d+1) \left(1.914 + \log_2 \frac{n}{d+1} \right) - 1$$

Proof. It follows from Theorem 1 that

$$B_{n,d} \leq t-1+d+(d+1)\lfloor \log_2 \frac{n}{t} \rfloor \leq t-1+d+(d+1) \log_2 \frac{n}{t}$$

holds for any integer $t \leq n$. For a given t, consider a real $x \in [t-1, t]$. There holds

$$t-1+d+(d+1) \log_2 \frac{n}{t} \leq \left(x - 1 + d + (d+1) \log_2 \frac{n}{x} \right) + 1,$$

because the derivative with respect to x of the expression between parentheses is bounded by 1 and $x - t \leq 1$. Besides, for every $x \leq n$. one can find an integer $t \leq n$ such that $x \in [t-1, t]$ by taking $t = \lceil x \rceil$. We have thus

$$B_{n,d} \leq x + d + (d+1) \log_2 \frac{n}{x}. \qquad (3)$$

The right-hand side expression reaches its minimum at $x^* = \frac{d+1}{\ln 2}$, which is smaller than n since $d \leq n \ln 2 - 1$. Reintroducing this in (3) leads after a few manipulations to the bound (2). □

Colored graphs

Our model allows for colored nodes, and the bounds that we have obtained are thus also valid for colored graphs. However, they do not take advantage of the possible presence of colors. The next corollary explicitly uses the colors of the nodes, and establishes a bound that can be much stronger than those of Theorem 1 and Corollary 1 when a color is shared by a small number of nodes. It is particularly relevant when a small set of particular nodes are "marked" or have a special property, as in [8], where some agents exploring the graph each have a homebase, and homebases are marked with special color.

Corollary 2 (Colored graphs)**.** *Let* G *be a connected colored graph of diameter* d*, and* C *the set of nodes of* G *having a certain color. Independently of* n*, two nodes have the same view if and only if they have the same view of depth* $d + (d+1)\lfloor \log_2 |C| \rfloor$*, where* $|C|$ *is the cardinality of* C*.*

Proof. Remember that the view of depth 0 contains the color of the node. The partition π_0 contains thus one block for each color, and in particular a block of size $|C|$ corresponding to the nodes of C. Lemma 4 implies then that all blocks of π_d have a size smaller than or equal to $|C|$. The result is then obtained by combining Lemmas 1 and 3 exactly as in the proof of Theorem 1, skipping Claim 1. □

Directed graphs

Our results have so far been stated for bidirectionnal graphs. They can be extended to some classes of graphs involving directed edges[1]: Indeed, our approach only relies on Lemmas 1, 2 and 3, as subsequent results are solely built on these lemmas and the (strong) connectivity of the graph. Lemma 1 is valid independently of the presence of directed edges, and Lemma 2 remains valid in the presence of directed edges provided that locally unique port numbers are also assigned to outgoing edges as in Figure 2(a). More formally, the injection σ_v defining the port numbers for every node should be defined on the set containing all the outgoing edges leaving v and the bidirectional edges incident to v.

The case of Lemma 2 is slightly more subtle, and depends on the way directed edges are taken into account in the view. To preserve its validity, there must be a correspondence between a directed path from w to v in the graph and a path in the view $T(v)$ of v from a node representing w to the root of the view (for any v, w). This is the case if the recursion used to define the view $T^k(v)$ from the views $T^{k-1}(v_i)$ of its neighbors, one includes all bidirectional edges incident to v *and all directed edges arriving at* v*,* as in the example in Figure 2(b).

If views are defined in this way and locally unique port numbers are assigned to every bidirectional edge and directed edge *leaving* a node as in Figure 2(a), then one can verify that all our proofs and results remain valid for strongly connected graphs. By symmetry, they also remain valid if views are defined using the outgoing directed edges as opposed the incoming ones, provided that locally unique port numbers are assigned to every every bidirectional edge and directed edge *arriving* at a node.

They are however not valid if the views are defined using the edges leaving the nodes but locally unique port numbers are assigned to outgoing and bidirectional edges, as can be seen in the example in Figure 2(c).

[1]In the context of views and computation on anonymous networks, a bidirectional edge is not necessarily equivalent to a pair of opposite directed edges. A node v connected to w by an incoming directed edge $\overrightarrow{(v,w)}$ and an outgoing directed edge $\overrightarrow{(w,v)}$ may indeed not know that the source of $\overrightarrow{(w,v)}$ is the same node as the target of $\overrightarrow{(v,w)}$.

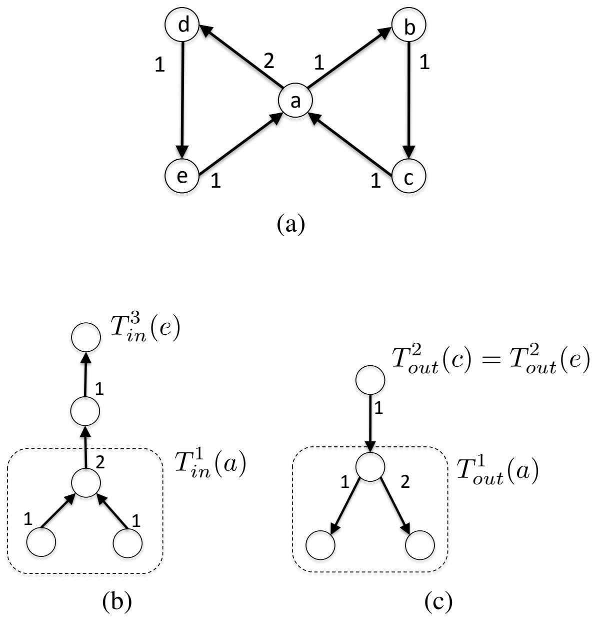


Figure 2. (a) Example of strongly connected graph with locally unique port numbers on the *outgoing edges*. (b) Construction of a view taking into account the directed edges *arriving* at the nodes. The view of depth 1 of node a is unique because it is the only node with in-degree 2. The view of depth 3 of node e is as a result unique because it is the only node reachable from a by a path with port sequence $(2, 1)$, consistently with Lemma 4. (c) Construction of a view taking into account the directed edges *leaving* the nodes. Lemma 4 is not valid in that case: the view of depth 1 of node a is shared by no other node since it is the only node with out-degree 2. However, one can verify that node c and e share the same view for every depth, as shown in (c) for a depth 2.

IV. Discussion

A. Implications of the new bound

We have shown that $B_{n,d} = O(d + d\log(n/d))$ while the best previously available bound was $B_{n,d} \leq n - 1$. The magnitude of the improvement provided by our bound depends on the relation between the diameter and the number of nodes. It is linear (at best) when $d = \Theta(n)$, but can be dramatic for networks with small diameter. In particular, when $d = O(\log n)$, we have $B_{n,d} = O(\log n + \log n \log \frac{n}{\log n}) = O(\log^2 n)$.

This improvement automatically impacts several bounds in the literature that rely on Norris's result. It allows for example constructing views of depth $O(d+d\log(n/d))$ instead of $n-1$ in the algorithm proposed in [12] to compute boolean functions on an anonymous network. More importantly, Norris's bound had been used to prove that the view of depth $(n-1)+(d+1)$ (or $2n-1$ when no other bound on the diameter than $n-1$ is available) contains all the information that can possibly be obtained by the nodes in the network by deterministic anonymous algorithms. In particular, it contains enough information to build all further views, and to compute any function that can be computed by a deterministic anonymous algorithm, without requiring any further communication. As a result, several algorithms in the literature involve building the views of depth $(n-1) + (d+1)$ or $2n-1$.

Yamashita et al. apply for example this idea to the election of a leader or an edge, the recovery of the network topology and the determination of a spanning tree [16]. Das et al. use a similar construction in an algorithm allowing mobile agents to meet on one node [8], and the idea is also used bu Dereniowski and Pelc for drawing maps of networks [9]. Besides, a variaton of the idea involving the concept of "fibrations" of graphs which is related to views has been used by Boldi et al. [3] to build a universal self-stabilizing algorithm, i.e. an algorithm that can self-stabilize on any behavior for which a self-stabilizing algorithm exists.

When our result applies and information on the diameter is available, our new bound on $B_{n,d}$ can be directly substituted for $n-1$, so that it is sufficient to build the views of depth $B_{n,d}+(d+1) = O(d+d\log(n/d))$ instead of $(n-1)+(d+1)$ in all these works.

Moreover, the gain in computation time and communication cost can even be stronger. Indeed, these costs grow both quadratically with the depth h of the view that nodes want to build if Tani's algorithm is used [14], which is the most efficient one of which we are aware in the context of distributed computation by nodes. (More specifically, they grow respectively as $O(h^2 n \log^2(n)\Delta^2)$ and $O(h^2 nm\Delta \log \Delta)$ if $h = O(n)$, where m is the number of edges, and Δ the maximal degree, see [14] and in particular Theorem 7 for more detail).

Our bound also decreases the cost of other types of algorithms. The algorithm of Andot et al. [1], designed to minimize the space complexity of leader election in an anonymous network, requires for example storing a constant number of paths of length $2n-1$, leading to a memory use of $O(n \log \Delta)$ (where Δ is the largest degree). The value $2n-1$ comes again from the depth at which the view of a node contains all the information avaible in the network, and can be substituted by $d+1+B_{n,d} = O(d+d\log(n/d))$ as above, leading to a total memory use of $O((d + d\log(n/d)) \log \Delta)$.

Chalopin et al. [4] have also proposed a method for constructing a map of an initially unknown network explored by an agent. Their agent follows a path defined by a special sequence of ports that guarantees that the agents passes by every edge in the network (assuming a bound on n and the degree are known), and explores parts of the views of every nodes that it encounters for different depths. This results in a procedure taking $O(k.n\Delta\,|U_{n,d}|)$ steps that must be iterated for every k smaller than $n-1$, where $|U_{n,d}|$ the length of the sequence of ports defining the path The bound $n-1$ on k comes from the use of Norris's bound, and can be substituted by $B_{n,d}$. Our results allow then having a total cost of $O(n(d + d\log(n/d))^2\Delta\,|U_{n,d}|)$ instead of $O(n^3\Delta\,|U_{n,d}|)$.

B. Tightness

Unlike Norris's bound which is tight when one only considers n, our bound could at least be marginally improved. Indeed, one could in principle obtain a stronger bound by computing, for every couple (n, d), the length of longest sequence of different partitions $\pi_0, \pi_1, \ldots$ consistent with the constraints imposed by Lemmas 1 and 4. Our result in

Theorem 1 only provides upper bounds on the solution to this combinatorial optimization problem, and a full analysis of some simple cases has shown that these family of bounds are not always tight. For example, solving the combinatorial optimization problem for $n = 9$ and $d = 1$ leads to the bound $B_{n,d} \leq 4$, while Theorem 1 only shows that $B_{n,d} \leq 5$, using $t = 3$.

Besides, as mentioned in the Introduction, Fraigniaud and Pelc [11] have shown a bound $\hat{n} - 1$, where $\hat{n}$ is the number of different views present in the network. This $\hat{n}$ is also the number of nodes in the "quotient graph" $\hat{G}$ of G, which is the smallest (multi)-graph generating the same set of views as G (It can be proved that the ratio $n/\hat{n}$ is always an integer [16]). An intuitive way of seeing this is that nodes or agents cannot distinguish G from $\hat{G}$, so that bounds that apply to $\hat{G}$ also apply to G. A similar argument applies to our results: It can be shown that they remain valid for the quotient graph, even though some care is needed to treat possible multiple edges and self-loops that were not present in our initial model. Then, since agents or nodes cannot distinguish $\hat{G}$ from G, the bounds that apply to the former also apply again to the latter. As a result, n and d can be replaced by $\hat{n}$ and $\hat{d}$, the number of nodes and diameter of the quotient graph, in all our bounds. However, using these stronger bounds requires having information about the quotient graph, which may not often be available in a decentralized context.

C. Diameter and actual value of $B_{n,d}$.

The diameter d is obviously a lower bound on $B_{n,d}$. On the other hand, we were so far not able to find graphs for which one would need to go at a depth larger[2] than $d + 1$ to find the final partition, and the example for which Norris's bound $n-1$ is tight has actually a diameter $d = n-1$. Norris has also shown an example of graph for which stopping at the depth d was not sufficient [13], but this graph does not fit in our framework, because several edges leaving the same node have the same label, so that a sequence of labels and a starting node do not necessarily define a unique path. (This is in particular the case for many of the paths realizing the diameter in that example).

In our framework, it is remarkable that views of depth $d + 1$ contain all the nodes and edges of the graph, together with paths to all these nodes and edges that can be uniquely specified by sequences of port numbers. Similarly, any view of length $2d + 1$ contains all the shortest paths of the graph. Maybe naively, we fail to see which additional information, not contained in views of depth $d + 1$ or $2d + 1$, would be contained in views of higher depth. We therefore wonder whether $B_{n,d} = O(d)$.

References

[1] E. Andot, H. Ono, K. Sadakane, and M. Yamashita. The space complexity of the leader election in anonymous networks. In *IEEE International Symposium on Parallel and Distributed Processing (IPDPS 2008)*, pages 1–8. IEEE, 2008.

[2] P. Boldi and S. Vigna. Fibrations of graphs. *Discrete Mathematics*, 243(1-3):21–66, 2002.

[3] P. Boldi and S. Vigna. Universal dynamic synchronous self–stabilization. *Distributed Computing*, 15(3):137–153, 2002.

[4] J. Chalopin, S. Das, and A. Kosowski. Constructing a map of an anonymous graph: Applications of universal sequences. *Principles of Distributed Systems*, pages 119–134, 2010.

[5] J. Chalopin, E. Godard, Y. Métivier, and R. Ossamy. Mobile agent algorithms versus message passing algorithms. *Principles of Distributed Systems*, pages 187–201, 2006.

[6] J. Chalopin and Y. Métivier. An efficient message passing election algorithm based on Mazurkiewicz's algorithm. *Fundamenta Informaticae*, 80(1):221–246, 2007.

[7] J. Czyzowicz, A. Kosowski, and A. Pelc. How to meet when you forget: log-space rendezvous in arbitrary graphs. In *Proceeding of the 29th ACM SIGACT-SIGOPS symposium on Principles of distributed computing*, pages 450–459. ACM, 2010.

[8] S. Das, M. Mihalák, R. Šrámek, E. Vicari, and P. Widmayer. Rendezvous of mobile agents when tokens fail anytime. *Principles of Distributed Systems*, pages 463–480, 2008.

[9] D. Dereniowski and A. Pelc. Drawing maps with advice. *Journal of Parallel and Distributed Computing*, 2011.

[10] D. Dereniowski and A. Pelc. Leader election for anonymous asynchronous agents in arbitrary networks. *arXiv preprint arXiv:1205.6249*, 2012.

[11] P. Fraigniaud and A. Pelc. Decidability classes for mobile agents computing. *LATIN 2012: Theoretical Informatics*, pages 362–374, 2012.

[12] E. Kranakis, D. Krizanc, and J. Vandenberg. Computing boolean functions on anonymous networks. *Information and Computation*, 114(2):214–236, 1994.

[13] N. Norris. Universal covers of graphs: Isomorphism to depth n-1 implies isomorphism to all depths. *Discrete Applied Mathematics*, 56(1):61–74, 1995.

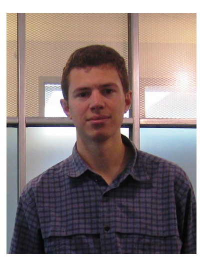
[14] S. Tani. Compression of view on anonymous networks -folded view-. *IEEE Transactions on Parallel and Distributed Systems*, 23(2):255–262, 2012.

[15] M. Yamashita and T. Kameda. Computing functions on asynchronous anonymous networks. *Theory of Computing Systems*, 29(4):331–356, 1996.

[16] M. Yamashita and T. Kameda. Computing on anonymous networks: Part I - Characterizing the solvable cases. *IEEE Transactions on Parallel and Distributed Systems*, 7(1):69–89, 1996.

[17] M. Yamashita and T. Kameda. Computing on anonymous networks: Part II - Decision and membership problems. *IEEE Transactions on Parallel and Distributed systems*, 7(1):90–96, 1996.

[2] Actually, in the the examples for which a depth $d + 1$ is needed, a depth d would have sufficed if views were defined to also include the degree of the leaves, i.e. if the view of depth 0 was defined to also include the degree of the node.



Julien M. Hendrickx received an engineering degree in applied mathematics and a PhD in mathematical engineering from the Université catholique de Louvain, Belgium, in 2004 and 2008, respectively.

He has been a visiting researcher at the University of Illinois at Urbana Champaign in 2003-2004, at the National ICT Australia in 2005 and 2006, and at the Massachusetts Institute of Technology in 2006 and 2008. He was a postdoctoral fellow at the Laboratory for Information and Decision Systems of the Massachusetts Institute of Technology 2009 and 2010, holding postdoctoral fellowships of the F.R.S.-FNRS (Fund for Scientific Research) and of Belgian American Education Foundation. Since September 2010, he is assistant professor (chargé de cours) at the Université catholique de Louvain, in the Ecole Polytechnique de Louvain.

Doctor Hendrickx is the recipient of the 2008 EECI award for the best PhD thesis in Europe in the field of Embedded and Networked Control, and of the Alcatel-Lucent-Bell 2009 award for a PhD thesis on original new concepts or application in the domain of information or communication technologies.