

Time Complexity of Consensus in Dynamic Networks Under Oblivious Message Adversaries

AMI PAZ, LISN – CNRS & Université Paris-Saclay, France

HUGO RINCON GALEANA, Technische Universität Wien, Austria

STEFAN SCHMID, Technische Universität Berlin, Germany & Universität Wien, Austria

ULRICH SCHMID, Technische Universität Wien, Austria

KYRILL WINKLER, Universität Wien, Austria

Consensus is a most fundamental task in distributed computing. This paper studies the consensus problem for a set of processes connected by a dynamic directed network, in which computation and communication is lock-step synchronous but controlled by an oblivious message adversary. In this basic model, determining consensus solvability and designing consensus algorithms in the case where it is possible, has been shown to be surprisingly difficult. We present an explicit decision procedure to determine if consensus is possible under a given adversary. This in turn enables us, for the first time, to study the time complexity of consensus in this model. In particular, we derive time complexity upper bounds for consensus solvability both for a centralized decision procedure as well as for solving distributed consensus. We complement these results with time complexity lower bounds. Intriguingly, we find that reaching consensus under an oblivious message adversary can take exponentially longer than broadcasting the input value of some process to all other processes.

CCS Concepts: • **Theory of computation** → **Distributed algorithms**; • **Networks**;

Additional Key Words and Phrases: consensus, distributed computing, time complexity, message adversary

1 INTRODUCTION

Consensus, a task in which multiple processes need to agree on some value, based on local inputs, is a fundamental problem in distributed computing. At the heart of this problem lies the question of whether and how it is possible for the processes to exchange enough information with each other in order to reach agreement, e.g., on a numerical value or on performing a joint action. While consensus has been studied intensively for several decades already, in many models of distributed computing, it is still unknown whether and how quickly consensus can be achieved.

This paper studies deterministic consensus in dynamic directed networks. The study of such networks is of both practical and theoretical interest. It is of *practical* relevance as the communication topology of many large-scale distributed systems is *dynamic* (e.g., due to mobility, interference, or failures) and its links often *asymmetric* (e.g., in optical or in wireless networks) [25]. It is also of fundamental *theoretical* interest, as solving consensus in dynamic directed networks is considered significantly more difficult [30, 31] than solving consensus in dynamic networks with bidirectional links [24].

We consider a worst-case perspective and assume that the information flow between the processes is controlled by an adversary. In particular, we study a lock-step synchronous model, where a *message adversary* [2] may drop an arbitrary set of messages sent by some processes in each round. This results in a sequence of directed communication graphs, whose edges tell which process can successfully send a message to which other process in a given round. We specifically

Authors' addresses: Ami Paz, LISN – CNRS & Université Paris-Saclay, France, ami.paz@lisn.fr; Hugo Rincon Galeana, Technische Universität Wien, ECS Group, Vienna, Austria, hugorincongaleana@gmail.com; Stefan Schmid, Technische Universität Berlin, Germany & Universität Wien, Faculty IV, Vienna, Austria, stefan.schmid@tu-berlin.de; Ulrich Schmid, Technische Universität Wien, ECS Group, Vienna, Austria, s@ecs.tuwien.ac.at; Kyrill Winkler, Universität Wien, Faculty of Computer Science, Vienna, Austria, kyrill.winkler@gmail.com.

consider the fundamental oblivious message adversary model introduced by Coulouma, Godard and Peters [12]. In this model, the adversary is represented by a set \mathbf{D} of allowed communication graphs, from which the adversary can pick one arbitrarily in each round.

The oblivious message adversary model is appealing because it is conceptually simple and still provides a highly dynamic network model: The set of allowed graphs can be arbitrary, and the nodes that can communicate with one another can vary greatly from one round to the next. It is hence also well-suited for settings where significant transient message loss occurs, such as in wireless networks subject to interference. Furthermore, this model includes as a special case the classic link failure model by Santoro and Widmayer [28], where up to f links may fail in each round: the model is equivalent to a set of allowed graphs which contains all communication graphs where $\leq f$ edges are missing.

Interestingly, determining consensus solvability for a given set of graphs \mathbf{D} and, in particular, designing a consensus algorithm which succeeds whenever this is possible, is difficult [12]. For example, sometimes a “weaker adversary”, i.e., an adversary that allows for more communication overall (e.g., supporting a larger set \mathbf{D} and failing less links), may render consensus impossible, while it is possible for a smaller set \mathbf{D} .

In this paper, we are primarily interested in the *time complexity* of consensus under oblivious message adversaries. Our work hence complements previous work, which either primarily focuses on the *feasibility* of consensus [12] or the simpler broadcast problem [16, 35]: how long it takes until the input value of some process has reached every other process.

1.1 Our Contributions

We consider the fundamental problem of distributed consensus in dynamic directed networks. In particular, we chart a landscape of the time complexity of consensus in the presence of oblivious message adversaries.

Our main technical contribution is an explicit decision procedure for deciding the solvability of deterministic consensus and its analysis. This allows us, for the first time, to study the time complexity of distributed consensus under oblivious message adversaries. In particular, we present time complexity upper bounds for consensus solvability both for a centralized decision procedure as well as for solving distributed consensus. We further complement these upper bounds with time complexity lower bounds.

Our results also shed an interesting new light on the relationship between distributed consensus and broadcast: as the input value of some process is known to reach all other processes in almost linear time under any oblivious message adversary [16], one might be tempted to expect that consensus solvability can also be decided fast. Our results show that, quite on the contrary, reaching consensus can take exponential time.

1.2 Related Work

Consensus is a fundamental task in distributed computing, and the question if and when consensus is possible has fascinated researchers at least since the influential impossibility result by Fischer, Lynch, and Paterson [14] and its generalizations [8]. Consensus problems come in different flavors and arise in many settings, including shared memory architectures, message-passing systems, and blockchains, among others [1, 9, 22, 27, 32].

Research on deterministic consensus in synchronous message-passing systems subject to link failures dates back to the seminal paper by Santoro and Widmayer [28], who showed that consensus is impossible if up to $n - 1$ messages may be lost each round. This result has later been generalized along many dimensions [7, 10–12, 15, 29, 30]. For example, in [30], Schmid et al. showed that consensus can even be solved when a quadratic number of messages is lost per round, provided these losses do not isolate the processes. Several generalized models have been proposed in the

literature [11, 17, 19], like the heard-of model by Charron-Bost and Schiper [11], and also different agreement problems like approximate and asymptotic consensus have been studied in these models [10, 15]. In many of these and similar works on consensus [5, 6, 9, 13, 26, 31, 34], a model is considered in which, in each round, a digraph is picked from a set of possible communication graphs. Afek and Gafni coined the term message adversary for this abstraction [2], and used it for relating problems solvable in wait-free read-write shared memory systems to those solvable in message-passing systems. For a detailed overview of the field, we refer to the recent survey by Winkler and Schmid [32].

An interesting alternative model for dynamic networks assumes a T -interval connectivity guarantee, that is, a common subgraph in the communication graphs of every T consecutive rounds [23, 24]. In contrast to our directional model, solving consensus is relatively simple here, since the T -interval connectivity model relies on bidirectional links and always connected communication graphs. For example, 1-interval-connectivity, the weakest form of T -interval connectivity, implies that all nodes are able to reach all the other nodes in the system.

Another related model arises in the context of wait-free computation in shared memory systems with immediate atomic snapshots. Roughly speaking, these systems can be described using one specific oblivious message adversary, containing all transitively closed tournaments. Wait-free computation in this context is often studied using topological tools [3, 4, 18, 20, 21]. This line of work did not provide any time complexity bounds for consensus in our model, however.

Closely related to our work is the paper by Coulouma, Godard, and Peters [12], who substantially refined the results of [29]. The authors consider oblivious message adversaries and identify an equivalence relation on the sets of communication graphs, which captures the essence of consensus impossibility via non-broadcastability of one of the equivalence classes (“beta classes”) of this relation. The paper also presents a distributed consensus algorithm that, essentially, computes the beta classes. However, in contrast to our paper, the main focus of this work is on feasibility of consensus.

To the best of our knowledge, we are the first to provide an efficient (centralized) decision procedure and a distributed consensus algorithm with worst-case time complexity guarantees under oblivious message adversaries.

1.3 Organization

The remainder of this paper is organized as follows. We introduce our formal model and terminology in Section 2. The description and analysis of our decision procedure and our consensus algorithm are presented in Section 3 and Section 4, respectively, and our lower bound results are presented in Section 5. We conclude our contribution and discuss directions for future work in Section 7. Due to space constraints, most proofs and additional findings are deferred to the appendix.

2 MODEL AND PRELIMINARIES

We assume a set $\Pi = \{p_1, \dots, p_n\}$ of n processes, which execute a deterministic distributed protocol to reach consensus. Processes operate in lock-step synchronous rounds, where each round consists of a phase of message exchanges among the processes, followed by some local computation, whose execution time is assumed to be negligible. We consider a *full information* protocol where, in each round, every process broadcasts its complete local history (its *view* obtained at the end of the previous round, or the initial state), and computes a deterministic *decision function* Δ based on its current view, which also involves all views it received from other processes in this round.

Each phase of message exchange is restricted by a (possibly different) directed graph on Π , called a *communication graph*, which is controlled by a message adversary. A message from p to q may be delivered in round r only if the

communication graph of round r contains the edge (p, q) . Since every process obviously knows its own current view, we just assume that the communication graph always contains all the self-loops. We use $\text{In}_G(v)$ denote the in-neighborhood of process v in a graph G . Messages are unacknowledged and rounds are communication-closed, i.e., messages that are sent in round r arrive in round r or not at all.

A *communication pattern* is a sequence of such communication graphs, which (along with the initial views of all processes and the decision function Δ) will uniquely define a run of the system. In the oblivious message adversary model, there is a set \mathbf{D} of allowed communication graphs, and the admissible communication patterns are all sequences of graphs from \mathbf{D} . For brevity, we identify our message adversary with its set of allowed communication graphs.

For a communication graph G , let $G^r = (G)_{i=1}^r$ denote the communication pattern that consists of r repetitions of G . For a set of communication graphs \mathbf{G} , let $\mathbf{G}^r = \{(G_i)_{i=1}^r : G_i \in \mathbf{G}\}$ be the set of communication patterns of length r that consist only of graphs from \mathbf{G} . Given a set of allowed graphs \mathbf{D} , the oblivious message adversary generated by \mathbf{D} may thus be written as \mathbf{D}^ω (ω denotes infinitely many repetitions of elements of \mathbf{D}).

Let $\sigma = (G_i)_{i=1}^r$ be a communication pattern, where its length $r \geq 1$ can be any integer or infinite (denoted ω), and let Σ be a set of communication patterns. We use $\sigma|_{r'} = (G_i)_{i=1}^{r'}$ to denote the r' -round prefix of σ , which is only defined if the length of σ is at least r' , and $\Sigma|_{r'} = \{\sigma|_{r'} : \sigma \in \Sigma\}$ to denote the set of all r' -round prefixes of Σ ; by convention, $\sigma|_0 = \varepsilon$, where ε is the empty word. We use $\sigma(r') = G_{r'}$ to denote the r' th graph of σ and $\Sigma(r') = \{\sigma(r') : \sigma \in \Sigma\}$ for the set of communication patterns Σ . If σ has a finite length r and H is an arbitrary communication graph, we write $\sigma' = \sigma \circ H$ to denote σ extended by H , i.e., the communication pattern of length $r + 1$ with $\sigma'(i) = \sigma(i)$ for all $i \leq r$ and $\sigma'(r + 1) = H$.

A *root component* of a graph is a strongly connected component that has no incoming edge from a node outside of the component. We call a graph G *rooted* if it has a single root component and write $\text{Root}(G)$ for the node set of the root component of G . Note that if a graph G is rooted then a node (in our context: a process) $p \in V(G)$ has a path to every other node (process) in G if and only if $p \in \text{Root}(G)$. In Claim 3 below, we show that consensus is trivially impossible if the set of allowed graphs contains a graph that is not rooted, and for this reason we consider adversaries whose set \mathbf{D} consists of rooted graphs only. A set of communication graphs \mathbf{S} is *root-compatible* if all their root components contain a common node, i.e., $\bigcap_{G \in \mathbf{S}} \text{Root}(G) \neq \emptyset$. We will show that root-compatibility is a central concept when it comes to consensus solvability.

In our full information protocol, the view of process p in σ at time (= end of round) $r \geq 1$ comprises the view of all the processes that p had in its in-neighborhood in the round r communication graph $\sigma(r)$, along with the round number r . The initial view of process p consists of its input value x_p (see the specification of the consensus problem below) and the round number 0. Formally, views are recursively defined as $\text{view}_\sigma(p, 0) = \{(p, 0, x_p)\}$ and, for $r > 0$, $\text{view}_\sigma(p, r) = (p, r, V_\sigma(p, r - 1))$, where $V_\sigma(p, r - 1) = \{\text{view}_\sigma(q, r - 1) : (q, p) \in \sigma(r)\}$.

For notational simplicity, we will subsequently use the tuple (p, r) , called a *process-time node*, to refer to the view of process p at time r . We thus use $(p, r') \rightsquigarrow_\sigma (q, r)$ to denote that p at time $r' < r$ has influenced q at time r , which can be expressed formally by the existence of a sequence of processes $p = p_1, \dots, p_{r-r'+1} = q$ satisfying $\text{view}_\sigma(p_i, r' + i - 1) \in V_\sigma(p_{i+1}, r' + i - 1)$ for $1 \leq i \leq r - r'$. We say that p is a *broadcaster* in σ (or equivalently, that a communication pattern σ is *broadcastable* by p), if $(p, 0) \rightsquigarrow_\sigma (q, r)$ for some time r , for all $q \in \Pi$.

Two communication patterns σ and σ' of the same length are *indistinguishable* by a process p , denoted $\sigma \sim_p \sigma'$, if this process has the same view in σ and in σ' , *eventually* or *in each round* in case of infinite patterns. Formally, $\sigma \sim_p \sigma' \Leftrightarrow \text{view}_\sigma(p, r) = \text{view}_{\sigma'}(p, r)$ if σ and σ' are r -round patterns, and $\sigma \sim_p \sigma' \Leftrightarrow \text{view}_\sigma(p, r) = \text{view}_{\sigma'}(p, r)$

for all r if σ and σ' are infinite. We write $\sigma \sim \sigma'$ if $\sigma \sim_p \sigma'$ for some p . We also use $\sigma \not\sim_p \sigma' \Leftrightarrow \neg(\sigma \sim_p \sigma')$, and $\sigma \not\sim \sigma' \Leftrightarrow (\forall p \in \Pi : \sigma \not\sim_p \sigma')$.

Given a set Σ of communication patterns of the same length, we define its *indistinguishability graph* $I(\Sigma)$ as follows. The nodes of $I(\Sigma)$ are the communication patterns in Σ , and the two communication patterns $\sigma, \sigma' \in \Sigma$ are connected by an edge if $\sigma \sim \sigma'$, i.e., if they are indistinguishable for some process. We label each edge with the set of processes defining it, that is, we define an edge labeling function $\ell : E(I(\Sigma)) \rightarrow 2^\Pi$ by $\ell((\sigma, \sigma')) = \{p \in \Pi : \sigma \sim_p \sigma'\}$.

Our first simple, yet important insight is that root components can preserve indistinguishability. Consider two communication patterns σ, σ' that are indistinguishable for a set of processes $\ell((\sigma, \sigma'))$, and assume that there is an allowed graph $G \in \mathbf{D}$ such that $\text{Root}(G) \subseteq \ell((\sigma, \sigma'))$. Then, the communication patterns $\sigma \circ G$ and $\sigma' \circ G$ are also indistinguishable for the processes in $\text{Root}(G)$: in G , these processes only receive messages from other members of $\text{Root}(G)$, and so these extended communication patterns are still indistinguishable for them.

CLAIM 1. *Let \mathbf{D} be an oblivious message adversary, r be a round, and $e = (\sigma, \sigma')$ be an edge in $I(\mathbf{D}^r)$. For $r > 1$, the edge $(\sigma|_{r-1}, \sigma'|_{r-1})$ is in $I(\mathbf{D}^{r-1})$. Moreover, if there is a graph $G \in \mathbf{D}$ such that $\text{Root}(G) \subseteq \ell(e)$ then the edge $e' = (\sigma \circ G, \sigma' \circ G)$ is in $I(\mathbf{D}^{r+1})$ and its label $\ell(e')$ satisfies $\text{Root}(G) \subseteq \ell(e') \subseteq \ell(e)$.*

PROOF. If $r > 0$, for every $p \in \ell(e)$, the indistinguishability $\sigma \sim_p \sigma'$ also implies $\sigma|_{r-1} \sim_p \sigma'|_{r-1}$, so the edge $(\sigma|_{r-1}, \sigma'|_{r-1})$ is indeed in $I(\mathbf{D}^{r-1})$.

To prove the second part of our claim, consider any process $p \in \text{Root}(G)$. By the definition of a root component, we have $\text{In}_G(p) \subseteq \text{Root}(p)$, so each process q with $(q, r) \in \text{view}_{\sigma \circ G}(p, r+1)$, is in $\text{Root}(G)$, and satisfies $\text{view}_\sigma(q, r) = \text{view}_{\sigma'}(q, r)$, because $\text{Root}(G) \subseteq \ell(e)$. This immediately implies that $\text{view}_{\sigma \circ G}(p, r+1) = \text{view}_{\sigma' \circ G}(p, r+1)$ and thus the edge e' exists and $\text{Root}(G) \subseteq \ell(e')$. The last part, $\ell(e') \subseteq \ell(e)$, follows because if $\text{view}_{\sigma \circ G}(q, r+1) = \text{view}_{\sigma' \circ G}(q, r+1) = (q, r+1, V_\sigma(q, r))$ for some process q then $\text{view}_\sigma(q, r) = \text{view}_{\sigma'}(q, r)$, as, by definition, $\text{view}_\sigma(q, r) \in V_\sigma(q, r)$. \square

In the **consensus problem**, each process p has an input value $x_p \in V$, taken from some finite domain V , and an output value y_p , initialized to \perp , to which it can write irrevocably, i.e., only once. An algorithm solves consensus in our setting if it ensures that

- eventually, every process p decides, i.e., assigns $y_p \neq \perp$ (termination),
- if $y_p \neq \perp$ and $y_q \neq \perp$ then $y_p = y_q$ for all $p, q \in \Pi$ (agreement),
- if $y_p = v \neq \perp$ then there is a process $q \in \Pi$ such that $x_q = v$ (validity).

Since we will consider full information protocols only, our consensus algorithm is actually a collection of decision functions. For every $p \in \Pi$, the decision function Δ_p maps every possible $\text{view}_\sigma(p, r)$ to a decision value $y_p \in V \cup \{\perp\}$, such that $\Delta(\text{view}_\sigma(p, r)) \neq \perp$ implies $\Delta(\text{view}_\sigma(p, r')) = \Delta(\text{view}_\sigma(p, r))$ for every $r' \geq r$. The *configuration* C_σ^r of our system at the end of round r in σ , is the vector of the elements $(\text{view}_\sigma(p, r), \Delta(\text{view}_\sigma(p, r)))$, for all p , and the *run* (also called execution in the literature) corresponding to σ is the sequence $(C_\sigma^r)_{r \geq 0}$. In the oblivious message adversary model, a run is uniquely determined by the input value assignment contained in the initial views and the communication pattern since the algorithm is deterministic.

With these definitions in mind, we now state two properties of consensus under oblivious message-adversaries, which will be of central importance in this paper. We first observe that any valid decision value must be the input value of a broadcaster. The proof of the following claim uses the same argument as [33, Theorem 2].

CLAIM 2. *Let \mathbf{D} be an oblivious message adversary and let $\sigma \in \mathbf{D}^\omega$. If in some correct consensus algorithm, all processes decide v in a run with σ , then v is the input value of a broadcaster in σ .*

PROOF. By the termination condition, there is a round r such that in all runs with σ all processes decide by this round when running a given correct consensus algorithm. Suppose that there is a r -round run ε with communication pattern σ where all processes decide v even though no broadcaster in σ has input value v . We show that this leads to a contradiction to the assumed correctness of the consensus algorithm.

Let $P = \{i_1, \dots, i_k\}$ be the identifiers of those processes that start with input value v in ε . By the validity condition, $P \neq \emptyset$. Let ε_j denote the run that is the same as ε , except that the processes with identifiers i_1, \dots, i_j have an input value $\neq v$. We show by induction that some process decides v in ε_j for $0 \leq j \leq k$. Thus in the run ε_k some process decides v , even though no process has input v in this run, a contradiction to the validity condition of consensus.

The base of the induction $j = 0$ follows immediately because $\varepsilon \sim \varepsilon_0 = \varepsilon$.

For the step from j to $j + 1$, where $0 \leq j < k$, we observe that, because σ is not broadcastable for any process with an identifier from P , there is a process q such that $(p_{i_{j+1}}, 0) \not\rightsquigarrow (q, r)$. Since ε_j is identical to ε_{j+1} except for the input of $p_{i_{j+1}}$, we have $\varepsilon_j \sim_q \varepsilon_{j+1}$. As all processes decide by round r in ε_j , and because they decide v by hypothesis, q and, by agreement, all processes decide v in ε_{j+1} . \square

Our second observation is that every communication graph in the set of allowed graphs of an oblivious message adversary, under which consensus is solvable, must be rooted.

CLAIM 3. *If an oblivious message adversary contains, in its set of allowed graphs \mathbf{D} , a graph G that is not rooted, then consensus is impossible.*

PROOF. The pattern $\sigma = G^\omega$ may be played by the adversary even though it is not broadcastable by any process, thus the claim follows from Claim 2. \square

3 A DECISION PROCEDURE FOR CONSENSUS SOLVABILITY

In this section, we present a decision procedure for determining whether consensus is solvable under an oblivious message adversary with a set \mathbf{D} of allowed graphs. In a nutshell, our procedure revolves around the (undirected) indistinguishability graph $I(\mathbf{D})$, constructed from the given input set \mathbf{D} : the nodes of the indistinguishability graph represent the graphs of \mathbf{D} and the edges represent indistinguishability. Given $I(\mathbf{D})$, we create a sequence $\mathcal{N}_1 = I(\mathbf{D}), \mathcal{N}_2, \dots$ of refinements of $I(\mathbf{D})$, and use the last graph \mathcal{N}_{TD} to decide if consensus is solvable under the message adversary \mathbf{D} . Here, TD is the number of iterations of the decision procedure, that is, the time complexity of the algorithm. In some sense, our decision procedure can essentially be viewed as an explicit computation of the abstract beta classes (and their broadcastability), as introduced by Couloma et al. [12]. As an additional feature, it reveals a crucial and previously unknown relation between the number of iterations of the decision procedure under a given oblivious message adversary and the time complexity of distributed consensus.

More concretely, our approach, summarized in Algorithm 1, uses the fact that a graph whose root component is a subset of $\ell(e)$ is suitable for perpetuating the indistinguishability for at least some of the processes of $\ell(e)$ (according to Claim 1). The algorithm starts from the indistinguishability graph $\mathcal{N}_1 = I(\mathbf{D})$ of \mathbf{D} , where \mathbf{D} is viewed as a set of 1-round communication patterns: the nodes of $I(\mathbf{D})$ are the graphs of \mathbf{D} , and two graphs $G, G' \in \mathbf{D}$ are connected by an edge if there is a process p that has the same set of incoming edges in G and in G' . The algorithm then computes a sequence (\mathcal{N}_i) of graphs, using iterative refinement. To refine from \mathcal{N}_{i-1} to \mathcal{N}_i , it keeps all \mathcal{N}_{i-1} 's nodes, but only a subset of its edges (Line 9): an edge $e = (u, v)$ is kept (by adding it to the set E_i) if the connected component of e in \mathcal{N}_{i-1} contains a communication graph G such that $\text{Root}(G) \subseteq \ell((u, v))$ (Line 8).

This procedure continues until the set of edges does not change for two successive iterations, or until all remaining connected components are root-compatible, i.e., all its communication graphs have a common member in their respective root components. As we will see later in Theorems 1 and 2, the root-compatibility of the connected components of the refined indistinguishability graph is precisely what is required to make consensus solvable.

For the algorithm, we assume that all graphs of \mathbf{D} have a unique root component, as consensus is trivially impossible otherwise (Claim 3). Note that, for two communication graphs G, H , we have $\ell((G, H)) = \{p \in \Pi : G \sim_p H\} = \{p \in \Pi : \text{In}_G(p) = \text{In}_H(p)\}$.

The following corollary provides a concise statement of the rule according to which the decision procedure selects which edges to keep when refining $\mathcal{N}_{i-1}(\mathbf{D})$ into $\mathcal{N}_i(\mathbf{D})$.

COROLLARY 1. *Let $e = (A, B)$ be an edge of $\mathcal{N}_i(\mathbf{D})$, for $i > 1$. Then in $\mathcal{N}_{i-1}(\mathbf{D})$:*

- (1) *the edge $e = (A, B)$ exists, and*
- (2) *there exists a node G_e with $\text{Root}(G_e) \subseteq \ell(e)$, such that A, B and G_e are in the same connected component.*

PROOF. According to Algorithm 1, an edge $e = (A, B)$ can only persist in \mathcal{N}_i if it was already present in \mathcal{N}_{i-1} and there was a corresponding graph G_e with $\text{Root}(G_e) \subseteq \ell(e)$ connected to A and B in \mathcal{N}_{i-1} . \square

We observe that, in order for an edge e of the indistinguishability graph to be “protected” from being omitted by the decision procedure by Line 9 of Algorithm 1, there must exist a communication graph whose root component is a subset of the label of e . This motivates the following definition.

DEFINITION 1. *Given a set of allowed graphs \mathbf{D} , let E be a set of edges of $I(\mathbf{D})$ and $\mathbf{G} \subseteq \mathbf{D}$ be a set of communication graphs. We call E protected by \mathbf{G} if for every $e \in E$ there is a graph $G_e \in \mathbf{G}$ such that $\text{Root}(G_e) \subseteq \ell(e)$.*

The following upper bound on the number of iterations TD of the decision procedure exploits the maximum number of different labels of the edges of $I(\mathbf{D})$.

CLAIM 4. *The number of iterations of the decision procedure, TD, satisfies $\text{TD} \leq 2^n$.*

PROOF. For a set of communication graphs \mathbf{G} , let $\mathcal{N}_i[\mathbf{G}]$ denote the subgraph of \mathcal{N}_i induced by \mathbf{G} . According to Algorithm 1, there must exist a set of communication graphs $\mathbf{G} \subseteq \mathbf{D}$ such that $\mathcal{N}_i[\mathbf{G}]$ is connected and not root-compatible for all $i < \text{TD}$, whereas all connected components of \mathcal{N}_{TD} are root-compatible. That is, \mathbf{G} constitutes the

Input: A set of allowed graphs \mathbf{D}

Output: The refined indistinguishability graph \mathcal{N}_{TD} .
Consensus is solvable if and only if all connected components of \mathcal{N}_{TD} are root-compatible.

```

// Initialization:
1  $i \leftarrow 1$ 
2  $\mathcal{N}_1 \leftarrow I(\mathbf{D})$ 
// Iterative construction:
3 repeat
4    $i \leftarrow i + 1$ 
5    $E_i \leftarrow \emptyset$ 
6   foreach  $e \in E_{i-1}$  do
7     Let  $\mathbf{G}$  be the communication graphs reachable from
        $e$  in  $\mathcal{N}_{i-1}$ 
8     if  $\exists G \in \mathbf{G} : \text{Root}(G) \subseteq \ell(e)$  then
9        $E_i \leftarrow E_i \cup \{e\}$ 
10     $\mathcal{N}_i \leftarrow \langle \mathbf{D}, E_i \rangle$ 
11 until  $\mathcal{N}_i = \mathcal{N}_{i-1}$  or all connected components of  $\mathcal{N}_i$  are
       root-compatible
12 return  $\mathcal{N}_{i-1}$ 

```

Fig. 1. The consensus decision procedure. It iteratively constructs the refined indistinguishability graph \mathcal{N}_{TD} for a set of allowed graphs \mathbf{D} .

last connected component of $I(\mathbf{D})$ that had to be broken apart by the decision procedure in order to arrive at a graph \mathcal{N}_{TD} where all connected components are root-compatible.

Furthermore, for $1 < i < \text{TD}$, the set $C_i(\mathbf{G})$ of nodes reachable from \mathbf{G} in \mathcal{N}_i satisfies $|C_i(\mathbf{G})| < |C_{i-1}(\mathbf{G})|$. This is because, if the $(i-1)^{\text{th}}$ iteration of the decision procedure does not result in the removal of a node from $C_{i-1}(\mathbf{G})$, then a set of edges that connects $C_{i-1}(\mathbf{G})$ in \mathcal{N}_{i-1} is protected by the communication graphs of C_{i-1} ; hence, no node will be removed from $C_j(\mathbf{G})$ for any $j \geq i$. This cannot come to pass, however, because then the decision procedure would already have terminated after $i < \text{TD}$ iterations.

In addition, all edges e of the connected component of \mathbf{G} in \mathcal{N}_i that have the same label $\ell(e) = \lambda$ are removed during a single iteration of the decision procedure: If e is removed from the connected component of \mathbf{G} in \mathcal{N}_i , then there is no communication graph in $C_i(\mathbf{G})$ that protects e and so all edges with label λ are removed from the connected component of \mathbf{G} . We recall that every label is a nonempty subset of Π , thus there are at most $2^n - 1$ different labels. The claim follows because, as we have shown above, $|C_i(\mathbf{G})| < |C_{i-1}(\mathbf{G})|$; hence at least one edge is removed from the connected component of \mathbf{G} in \mathcal{N}_i during the i^{th} iteration of the decision procedure. \square

Before looking more closely into the ramifications of a large number of iterations TD of the decision procedure of a given oblivious message adversary \mathbf{D} , it is instructive to study a few “extreme” examples of such adversaries, and, in particular, how the number of communication graphs $|\mathbf{D}|$ relates to TD. First, one may wonder whether the decision procedure can be fast if the set \mathbf{D} of allowed graphs is exponentially large. An example for such a scenario, in which consensus is solvable, is the set of all communication graphs that consist of a single clique of a fixed size $\lfloor n/c \rfloor$, for a constant c , and all the edges from each clique node to all other nodes (plus the self loops). There are exponentially many such graphs, yet no two are indistinguishable to any of the nodes, so the decision procedure already terminates after the first iteration because all connected components in $I(\mathbf{D})$ consist of a single communication graph. An example where a fast decision is possible despite an exponentially sized \mathbf{D} , where consensus is impossible, is the set of all rooted trees for $n > 2$. In this case, there is a path in $I(\mathbf{D})$ connecting every two trees T_1, T_2 . Also, every edge e in $I(\mathbf{D})$ has a corresponding tree $T \in \mathbf{D}$ that protects this edge, since there is a tree T with $\text{Root}(T) \subseteq \ell(e)$.

Complementing these insights, the question arises whether there are examples where TD is (almost) the same as $|\mathbf{D}|$. We will answer this question affirmatively (in Section 5), by giving an explicit example where TD is even exponential in n . In a nutshell, we will choose a set of communication graphs $\mathbf{D} = \{G_1, \dots, G_{\text{TD}}\}$, where the root component of each graph consists of a different set of processes of the same cardinality, i.e., for every $G, G' \in \mathbf{D}$ we have $|\text{Root}(G)| = |\text{Root}(G')|$, but if $G \neq G'$ then $\text{Root}(G) \neq \text{Root}(G')$. Furthermore, we let

$$G_1 \sim_{R_3} G_2 \sim_{R_4} \dots \sim_{R_{\text{TD}}} G_{\text{TD}-1} \sim_S G_{\text{TD}}, \quad (1)$$

where $R_i = \text{Root}(G_i)$ and S is a nonempty set such that no $G \in \mathbf{D}$ satisfies $\text{Root}(G) \subseteq S$. Here, the decision procedure can remove only the rightmost edge \sim_S in the first iteration, only the edge $\sim_{R_{\text{TD}}}$ in the second iteration, and so on, because all the remaining edges are protected by one of the remaining graphs.

Also in this case, consensus might be solvable (as in the example in Section 5 described above), or it might be impossible, as in the instance

$$G'_1 \sim_{R'_3} G'_2 \sim_{R'_4} G'_3 = G_1 \sim_{R_3} G_2 \sim_{R_4} \dots \sim_{R_{\text{TD}}} G_{\text{TD}-1} \sim_S G_{\text{TD}}$$

where we assume that G'_1 and G'_2 are chosen such that they are not root-compatible: in this case, the indistinguishability $G'_1 \sim_{R'_3} G'_2$ will never break.

In view of the above results, it might be tempting to assume that TD also determines the termination time of distributed consensus. Interestingly, this is not the case. Complementing the result of Theorem 1 established in Section 4, we will show in Section 6 that there are instances of oblivious message adversaries where the decision procedure terminates after a constant number of iterations, while the consensus termination time is exponential in n .

4 TIME COMPLEXITY OF CONSENSUS

In this section, we study the time complexity of consensus, and also ascertain our claim from Section 3, namely, that the decision procedure of Algorithm 1 correctly assesses oblivious message adversaries where consensus is solvable. Thus, throughout this section, we consider an oblivious message adversary, where, after some number TD of iterations, Algorithm 1 determined that all connected components of the refined indistinguishability graph \mathcal{N}_{TD} are root-compatible.

For solving consensus, we use the fact that non-connectivity in \mathcal{N}_{TD} implies non-connectivity in $I(\mathbf{D}^{(n-1)\text{TD}+1})$, in the following sense: Let C_1 and C_2 be two different connected components of \mathcal{N}_{TD} , and $t > (n-1)\text{TD}$. Then, any two communication patterns $\sigma_1 \in C_1^t$ and $\sigma_2 \in C_2^t$, consisting only of graphs of C_1 and C_2 , respectively, are not connected in the indistinguishability graph $I(\mathbf{D}^t)$.

We then apply a pigeon-hole argument to show that all connected components of $I(\mathbf{D}^{ct})$ are broadcastable, where c is the number of connected components of \mathcal{N}_{TD} . Note that this choice guarantees that graphs from at least one connected component are used at least t times. From here, a consensus decision function Δ_p can be easily defined by (i) for each connected component C of $I(\mathbf{D}^{ct})$, choosing one of its broadcasters, denoted $b(C)$, and (ii) if p 's view is consistent with a graph sequence σ , and σ belongs to a connected component C of $I(\mathbf{D}^{ct})$, then p decides on the input $x_{b(C)}$ of $b(C)$, for which $\text{view}_\sigma(b(C), 0, x_{b(C)})$ must already be present in p 's view.

It is rather immediate that such a procedure solves consensus, given the mapping $b(C)$, which we will prove in the remainder of this section: Termination follows from the existence of the mapping $b(C)$; validity follows because the decided value was some process' input value; agreement is a consequence of all pairwise indistinguishable views lying in the same connected component C of $I(\mathbf{D}^{ct})$. Hence two different decisions can only occur in runs that are distinguishable for everyone (and are thus distinct runs).

A path $\pi = (\sigma_0, \dots, \sigma_s)$ in $I(\mathbf{D}^r)$ is a sequence of communication patterns such that $(\sigma_i, \sigma_{i+1}) \in E(I(\mathbf{D}^r))$ for all $0 \leq i < s$. Given such a path and $r' \leq r$, we write $\pi|_{r'}$ to denote the path $(\sigma_0|_{r'}, \dots, \sigma_\ell|_{r'})$ in $I(\mathbf{D}^{r'})$ of the r' -round prefixes of the communication patterns in π , which exists by Claim 1. Similarly, we denote by $\pi(r')$ the path $(\sigma_0(r'), \dots, \sigma_\ell(r'))$ in $I(\mathbf{D})$ of the r' th graphs of the communication patterns in π . Both $\pi|_{r'}$ and $\pi(r')$ are indeed paths in the corresponding indistinguishability graphs, due to a more general claim: removing an intermediate communication round from all communication patterns in a path cannot disconnect it, as stated below.

For a communication pattern σ of length r , and some round $r' \leq r$, let $\sigma - r'$ denote $\sigma|_{r'-1} \circ \sigma(r'+1) \circ \dots \circ \sigma(r)$, i.e., the communication pattern σ with the round r' communication graph omitted. Corollary 2 shows that edges, and hence paths, between communication patterns in $I(\mathbf{D}^r)$ are preserved when omitting some round r' .

COROLLARY 2. *If the edge (σ, σ') is in $I(\mathbf{D}^r)$, then the edge $(\sigma - r', \sigma' - r')$ is in $I(\mathbf{D}^{r-1})$ as well.*

PROOF. Assume for contradiction that the edge is not preserved, i.e., $\sigma \sim \sigma'$ while $\sigma - r' \not\sim \sigma' - r'$. So, there is a process p such that $\sigma \sim_p \sigma'$ (this is true for at least one process, p) while $\sigma - r' \not\sim_p \sigma' - r'$ (this is true for all processes, and specifically for p). This implies that there exists a round $r'' \neq r'$ and a process q with w.l.o.g. $(q, r'') \rightsquigarrow_{\sigma - r'} (p, r)$ but $(q, r'') \not\rightsquigarrow_{\sigma' - r'} (p, r)$ or $\text{view}_{\sigma - r'}(q, r'') \neq \text{view}_{\sigma' - r'}(q, r'')$: if no such q, r'' existed, we would have $\sigma - r' \sim_p \sigma' - r'$.

Since $(q, r'') \rightsquigarrow_{\sigma-r'} (p, r)$, we also have $(q, r'') \rightsquigarrow_{\sigma} (p, r)$, as the sequence of processes causing (q, r'') to be in $\text{view}_{\sigma-r'}(p, r)$ also exists in σ and we just need to take path where the process of round r' is the same as of round $r' - 1$. To finish, it suffices to consider two cases: if $(q, r'') \not\rightsquigarrow_{\sigma'} (p, r)$, then p distinguishes σ and σ' since it has $\text{view}_{\sigma}(q, r'')$ in its view in σ but does not have $\text{view}_{\sigma'}(q, r'')$ in its view in σ' ; if $(q, r'') \rightsquigarrow_{\sigma'} (p, r)$, then p distinguishes σ and σ' by having $\text{view}_{\sigma}(q, r'') \neq \text{view}_{\sigma'}(q, r'')$ in its views. In both cases $\sigma \not\sim_p \sigma'$, a contradiction. \square

The following corollary relates the preservation of an edge in $I(\mathbf{D}')$ to the root components of the communication graphs that occur in the communication patterns of this edge.

COROLLARY 3. *Let \mathbf{D} be a set of allowed graphs and $0 < r' < r$ integers. Consider an edge $e = (\sigma, \sigma') \in I(\mathbf{D}')$ such that $e' = (\sigma|_{r'}, \sigma'|_{r'}) \in I(\mathbf{D}')$ satisfies $\sigma|_{r'} \neq \sigma'|_{r'}$. Then, there are at most $|\ell(e')| - 1$ rounds r_j , $r' < r_j \leq r$, satisfying $\text{Root}(\sigma(r_j)) \not\subseteq \ell(e')$.*

PROOF. By Claim 1, we can be sure that e' exists. For a contradiction, suppose that there are $|\ell(e')|$ rounds $r' < r_1 < \dots < r_{|\ell(e')|} \leq r$ such that each r_j satisfies $\text{Root}(\sigma(r_j)) \not\subseteq \ell(e')$. Let

$$U_j = \{p \in \Pi : \exists q \in \Pi \setminus \ell(e') (q, r') \rightsquigarrow_{\sigma} (p, r_j)\} \quad (2)$$

denote the set of processes that received a message by round r_j , sent after round r' , from a process outside of $\ell(e')$. Let $r_0 = r'$ and $U_0 = \Pi \setminus \ell(e')$. Note that from $\sigma|_{r'} \neq \sigma'|_{r'}$ it follows that $\emptyset \neq \ell(e') \neq \Pi$ and thus $U_0 \neq \emptyset$.

Let $\bar{U}_j = \Pi \setminus U_j$ and consider the cut (U_j, \bar{U}_j) in $\sigma(r_j)$, the communication graph at round r_j . Since we have $\text{Root}(\sigma(r_j)) \not\subseteq \ell(e')$, there is a process $p' \in \text{Root}(\sigma(r_j)) \setminus \ell(e')$. On the one hand, $p' \in \text{Root}(\sigma(r_j)) \setminus \ell(e')$ immediately implies $p' \in U_j$, since $(p', r') \rightsquigarrow_{\sigma} (p', r_j)$. On the other hand, $p' \in \text{Root}(\sigma(r_j))$ implies that in $\sigma(r_j)$ there is a path from p' to every node. Hence, if $\bar{U}_j \neq \emptyset$, then there is a node $p'' \in \bar{U}_j$, and a path in $\sigma(r_j)$ from p' to p'' ; this path must cross an edge \tilde{e}_j from U_j to \bar{U}_j .

We now use induction on $j = 0, \dots, |\ell(e')|$ to show that $|U_j| \geq n - |\ell(e')| + j$. For the basis $j = 0$, we have already shown that $|U_0| = n - |\ell(e')| > 0$. In the induction step, we prove that U_j grows by at least one (unless $U_j = \Pi$) due to the edge $\tilde{e}_j = (q', q'')$ from U_j to \bar{U}_j . As, for every $q \in \Pi \setminus \ell(e')$ in the definition of U_j in Eq. (2), $(q, r') \rightsquigarrow_{\sigma} (q', r_j)$ in conjunction with $(q', r_j) \rightsquigarrow_{\sigma} (q'', r_{j+1})$ implies $(q, r') \rightsquigarrow_{\sigma} (q'', r_{j+1})$, we obtain $U_{j+1} \supseteq U_j \cup \{q''\}$ as required.

It hence follows that $|U_{|\ell(e')|}| = n$, i.e., by round $r \geq r_{|\ell(e')|}$, every process has received a message, sent after round r' , from a process q outside of $\ell(e')$. Consequently, at time r , the view of every process contains the view of a process q that could distinguish $\sigma|_{r'}$ and $\sigma'|_{r'}$, hence every process can also distinguish σ and σ' . Formally, $\forall p \in \Pi \exists q \in \Pi \setminus \ell(e') : (q, r') \rightsquigarrow_{\sigma} (p, r)$ and $\text{view}_{\sigma}(q, r') \neq \text{view}_{\sigma'}(q, r')$, which implies that $\text{view}_{\sigma}(p, r) \neq \text{view}_{\sigma'}(p, r)$. That is, every process that can distinguish $\sigma|_{r'}$ and $\sigma'|_{r'}$ can also distinguish σ and σ' , contradicting the existence of the edge $e_r = (\sigma, \sigma')$ in $I(\mathbf{D}')$. \square

We proceed with Lemma 1, which generalizes and formalizes chains like Eq. (1), made up of connected subgraphs $\mathcal{S}_1, \dots, \mathcal{S}_i$ which are interconnected in a chain. It makes clever use of protected edges in order to delay the separation of root-incompatible connected components as much as possible, namely, by removing the interconnects between \mathcal{S}_j and \mathcal{S}_{j+1} in \mathcal{N}_{i-j} , i.e., from right (i) to left (1).

LEMMA 1. *Given a message adversary \mathbf{D} and i connected subgraphs $\mathcal{S}_1, \dots, \mathcal{S}_i$ of $I(\mathbf{D})$ such that for every $1 \leq j < i$, the edges of $\bigcup_{j'=1}^j \mathcal{S}_{j'}$ are protected by the communication graphs of $\bigcup_{j'=1}^{j+1} \mathcal{S}_{j'}$, and \mathcal{S}_j is connected to \mathcal{S}_{j+1} in \mathcal{N}_{i-j} , it holds that \mathcal{S}_1 is a connected subgraph of \mathcal{N}_i .*

PROOF. We show that all edges of \mathcal{S}_1 are in \mathcal{N}_i . In order to do so, we prove by induction on $i' = 1, \dots, i$, that all edges of $\bigcup_{j'=1}^{i-i'+1} \mathcal{S}_{j'}$ are in $\mathcal{N}_{i'}$.

The base $i' = 1$ follows directly from the code of Algorithm 1: $\mathcal{N}_1 = I(\mathbf{D})$, and each graph $\mathcal{S}_{j'}$ is a subgraph of $I(\mathbf{D})$, thus every edge of $\bigcup_{j'=1}^i \mathcal{S}_{j'}$ is in \mathcal{N}_1 .

For the inductive step from i' to $i' + 1$, assume that every edge of $\bigcup_{j'=1}^{i-i'+1} \mathcal{S}_{j'}$ is present in $\mathcal{N}_{i'}$. By assumption, every edge e of $\bigcup_{j'=1}^{i-i'} \mathcal{S}_{j'}$ is protected by a communication graph G of $\bigcup_{j'=1}^{i-i'+1} \mathcal{S}_{j'}$, i.e., by Definition 1, $\text{Root}(G) \subseteq \ell(e)$. As we also assume that \mathcal{S}_j is connected to \mathcal{S}_{j+1} in \mathcal{N}_{i-j} for $1 \leq j < i$, we have that $\mathcal{S}_{i-i'-j'}$ is connected to $\mathcal{S}_{i-i'-j'+1}$ in $\mathcal{N}_{i'+j'}$ for $0 \leq j' < i - i'$. Since $\mathcal{N}_{i'+j'}$ is a refinement of $\mathcal{N}_{i'}$, $\mathcal{S}_{i-i'-j'}$ is connected to $\mathcal{S}_{i-i'-j'+1}$ also in $\mathcal{N}_{i'}$. Hence $\bigcup_{j'=1}^{i-i'+1} \mathcal{S}_{j'}$ is a connected subgraph of $\mathcal{N}_{i'}$, and thus e is connected to G in $\mathcal{N}_{i'}$. Thus, in $\mathcal{N}_{i'}$, e is in the same connected component as a graph G with $\text{Root}(G) \subseteq \ell(e)$ and, by Line 8 of Algorithm 1, we have $e \in \mathcal{N}_{i'+1}$. \square

We are now ready to prove the main technical result of this section. For $r = (n-1) \cdot \text{TD}$, we show how the connectivity of two r -round communication patterns in $I(\mathbf{D}^r)$, consisting only of communication graphs from certain sets \mathbf{C}_1 and \mathbf{C}_2 , respectively, is related to the connectivity of \mathbf{C}_1 and \mathbf{C}_2 in the refined indistinguishability graph \mathcal{N}_{TD} , as computed by Algorithm 1.

LEMMA 2. *Given an oblivious message adversary \mathbf{D} , let \mathbf{C} constitute a connected component of \mathcal{N}_{TD} and let $\bar{\mathbf{C}} = \mathbf{D} \setminus \mathbf{C}$. For $r = (n-1) \cdot \text{TD}$, there is no connection in $I(\mathbf{D}^r)$ between any $\sigma_1 \in \mathbf{C}^r$ and any $\sigma_2 \in \bar{\mathbf{C}}\mathbf{D}^{r-1}$. Herein, $\sigma_2 \in \bar{\mathbf{C}}\mathbf{D}^{r-1}$ denotes the fact that σ_2 is composed of one graph of $\bar{\mathbf{C}}$ and then $r-1$ graphs of \mathbf{D} .*

PROOF. Assume for a contradiction that there exist $\sigma_1 \in \mathbf{C}^r$ and $\sigma_2 \in \bar{\mathbf{C}}\mathbf{D}^{r-1}$ which are connected in $I(\mathbf{D}^r)$. We show that \mathbf{C} is connected to some node of $\bar{\mathbf{C}}$ in \mathcal{N}_{TD} , contradicting the fact that \mathbf{C} is a connected component of \mathcal{N}_{TD} . We do so by proving that there are TD connected subgraphs $\pi_1, \dots, \pi_{\text{TD}}$ in $I(\mathbf{D})$, such that each of them intersects \mathbf{C} , π_1 also intersects $\bar{\mathbf{C}}$, and, for every $1 \leq j < i = \text{TD}$, the edges of $\bigcup_{j'=1}^j \pi_{j'}$ are protected by the communication graphs of $\bigcup_{j'=1}^{j+1} \pi_{j'}$. Moreover, π_j is connected to π_{j+1} in \mathcal{N}_{i-j} : We have that π_j and π_{j+1} both intersect \mathbf{C} , and since \mathbf{C} is a connected component in \mathcal{N}_i and \mathcal{N}_i is a refinement of \mathcal{N}_{i-j} , all nodes of \mathbf{C} are in the same connected component of \mathcal{N}_{i-j} . We can hence apply Lemma 1, which reveals that π_1 is a connected subgraph of \mathcal{N}_i . As π_1 also intersects both \mathbf{C} and $\bar{\mathbf{C}}$, however, we have the required contradiction.

Let $\tilde{\pi}$ be a path that connects σ_1 and σ_2 in $I(\mathbf{D}^r)$. Recall that, for a round $r' \leq r$, $\tilde{\pi}(r')$ denotes the round r' communication graphs $\sigma(r')$ for all communication patterns σ of $\tilde{\pi}$. By a repeated application of Corollary 2, we get that $\tilde{\pi}(r')$ is a path that connects $\sigma_1(r') \in \mathbf{C}$ and $\sigma_2(r') \in \mathbf{D}$ in $I(\mathbf{D})$ where, in particular, $\tilde{\pi}(1)$ connects $\sigma_1(1) \in \mathbf{C}$ and $\sigma_2(1) \in \bar{\mathbf{C}}$.

We now construct each connected subgraph π_j , $1 \leq j \leq i$, as a union of paths $\tilde{\pi}(r')$. That is, for some set $R_j \subseteq \{1, \dots, r\}$ of rounds, which we will define below, we set $\pi_j = \bigcup_{r' \in R_j} \tilde{\pi}(r')$. We denote the largest round of R_j as $r_j^* = \max(R_j)$.

For $1 \leq m < i$, we inductively construct R_{m+1} from R_m , starting with $R_1 = \{1\}$, i.e., setting $\pi_1 = \tilde{\pi}(1)$. We will assert that (1) $r_{m+1}^* \leq r_m^* + n - 1$ and (2) the edges of $\pi_m = \bigcup_{r' \in R_m} \tilde{\pi}(r')$ are protected by the communication graphs of $\pi_{m+1} = \bigcup_{r' \in R_{m+1}} \tilde{\pi}(r')$. For $1 \leq m \leq \text{TD}$, property (1) together with $r_1^* = 1$ guarantees $r_m^* \leq (n-1)(m-1) + 1 \leq (n-1) \cdot \text{TD} = r$, thus $\tilde{\pi}(r')$ is well-defined for all $r' \in R_m$.

Given R_m for $1 \leq m < i$, we construct R_{m+1} as follows: By Corollary 3, for every edge $e \in \pi_m$, there is a round $r_e \leq r_m^* + n - 1$ such that $\tilde{\pi}(r_e)$ contains a graph G with $\text{Root}(G) \subseteq \ell(e)$. Let R_{m+1} be the set of all such rounds, i.e., $R_{m+1} = \bigcup_{e \in E(\pi_m)} \tilde{\pi}(r_e)$. This ensures (1) by construction and also (2), because every edge e of π_m is protected by a

communication graph G of $\tilde{\pi}(r_e) \subseteq \pi_{m+1}$. Hence, the edges of π_m are protected by the communication graphs of π_{m+1} and so the edges of $\bigcup_{k=1}^m \pi_k$ are protected by the communication graphs of $\bigcup_{k=1}^{m+1} \pi_k$. \square

We are now ready to state the main theorem of this section, namely, an upper bound on the decision time complexity of consensus.

THEOREM 1. *Let \mathbf{D} be the set of allowed communication graphs of an oblivious message adversary. If the connected components of $\mathcal{N}_{\text{TD}}(\mathbf{D})$ are root-compatible, then consensus is solvable by round $c(n-1)(\text{TD}+1)$, where c is the number of connected components in \mathcal{N}_{TD} .*

PROOF. We show that every connected component of the indistinguishability graph $I(\mathbf{D}^t)$ is broadcastable for $t = c(n-1)(\text{TD}+1)$. This implies the theorem, because there exists a mapping for every connected component C of $I(\mathbf{D}^t)$ to a process p , such that p is a broadcaster in every communication pattern of C . More specifically, as C is an indistinguishability component, there is, for every process q and every $\sigma \in \mathbf{D}^t$, a map $\text{view}_\sigma(q, t) \mapsto p$ such that p is a broadcaster in every communication pattern of σ 's connected component in $I(\mathbf{D}^t)$. In every run with a communication pattern from C , every process has thus already learned the input x_p of p , which is a valid decision value. This decision procedure hence defines a correct consensus algorithm.

It remains to show the broadcastability of the connected components of $I(\mathbf{D}^t)$. Consider a run $\sigma \in \mathbf{D}^t$, and all the communication patterns $\sigma(i)$, $i = 1 \dots, c(n-1)(\text{TD}+1)$ appearing in it. By the pigeon-hole principle, at least one connected component C of \mathcal{N}_{TD} must supply $(n-1)(\text{TD}+1)$ of these graphs, when counted with repetitions. That is, there is a set $R \subseteq \{1, \dots, c(n-1)(\text{TD}+1)\}$, with $|R| = (n-1)(\text{TD}+1)$, such that every r_i with $i \in R$ satisfies $\sigma(r_i) \in C$. Note that the occurrence of $n-1$ or more graphs from C in σ already suffices to ensure that it is broadcastable by every process $p \in \bigcap_{G \in C} \text{Root}(G)$, i.e., that every process $q \in \Pi$ has $(p, 0, x_p) \in \text{view}_\sigma(q, t)$.

Consider another run $\sigma' \in \mathbf{D}^t$ that is connected to σ in $I(\mathbf{D}^t)$, and the communication patterns $\sigma'(i)$ appearing in it. If $n-1$ or more of the latter satisfied $\sigma'(r_i) \in C$, σ' would also be broadcastable by $\bigcap_{G \in C} \text{Root}(G)$, so assume that this is not the case. There are hence at most $n-2$ indices $r_j \in R$ where $\sigma'(r_j) \in C$. Let $R' \subseteq R$ with $|R'| = (n-1) \cdot \text{TD}$ be the set of indices obtained by discarding all these indices r_j from R , in addition to discarding some additional indices $\neq 1$ so as to match the desired size of R' .

We now construct the $((n-1) \text{TD})$ -round communication patterns ρ, ρ' defined by $\rho(j) = \sigma(r_j)$, $\rho'(j) = \sigma'(r_j)$ for each $j \in R'$. That is, starting out from σ and σ' , which are connected in $I(\mathbf{D}^t)$, we remove all communication rounds not in R' . By Corollary 2, ρ and ρ' are connected in $I(\mathbf{D}^{(n-1) \text{TD}})$. This, however, contradicts Lemma 2, because $\rho \in C^{(n-1) \text{TD}}$ and $\rho' \in \bar{C}^{(n-1) \text{TD}} \subseteq \bar{C} \times \mathbf{D}^{(n-1) \text{TD}-1}$ by construction, where C is a connected component in \mathcal{N}_{TD} and \bar{C} is its complement. \square

5 LOWER BOUNDS

This section complements our positive results above by studying lower bounds. In the following, we first establish a relationship between the time complexity of the decision procedure and the termination time of consensus. We will then derive a time complexity lower bound for the decision procedure, and combine it with the first result to establish a consensus termination time lower bound.

5.1 Decision complexity and consensus termination time

First, we present a relationship (Theorem 2) between the number of iterations of Algorithm 1 and the time complexity of consensus. As before, let $\mathcal{N}_i = \mathcal{N}_i(\mathbf{D})$ be the refined indistinguishability graph \mathcal{N}_i after i iterations according to Algorithm 1, with the set of allowed graphs \mathbf{D} sometimes omitted for brevity. Our general strategy is to establish that the impossibility of consensus after i rounds is equivalent to the existence of a set of “broadcast-incompatible” communication patterns of length i , which are connected to each other in the indistinguishability graph $I(\mathbf{D}^i)$. We ensure broadcast-incompatibility by letting this set also contain communication patterns G^i , i.e., i repetitions of the same communication graph G , taken from a set of root-incompatible graphs. Due to the requirement that every decision must be on the input of some broadcaster whose input value has reached everyone (recall Claim 2), this suffices: in G^i , the only processes that have reached everyone are the members of $\text{Root}(G)$, the root component of G . Thus, not all these communication patterns can have led to the same decision value, which is a contradiction since all connected round i communication patterns must have led to the same decision value if consensus was solved after i rounds.

The core of our proof is contained in Lemma 3. It shows that the connectivity of some communication graphs A, B in $\mathcal{N}_i(\mathbf{D})$ implies the connectivity of the communication patterns A^i, B^i in the indistinguishability graph $I(\mathbf{D}^i)$. Informally speaking, it uses an inductive construction for an arbitrary edge (A, B) of \mathcal{N}_i to show how the corresponding connectivity between A^i and B^i can be preserved for i rounds in $I(\mathbf{D}^i)$. It crucially relies on the fact that every \mathcal{N}_i is a refinement of \mathcal{N}_{i-1} , with \mathcal{N}_1 being a refinement of $I(\mathbf{D})$, which is due to the fact that Algorithm 1 iteratively only removes selected edges via Line 9 but never adds any edges.

To show that the connectivity of A^i and B^i is preserved, we use the path in \mathcal{N}_i from A to $\ell(e)$, respectively B to $\ell(e)$, to extend the already constructed connected prefixes A^{i-1} and B^{i-1} . Note that this path also occurs in \mathcal{N}_{i-1} due to Corollary 1. To illustrate this, consider a (very simple) example, where we have that $A \sim_p B$ occurs in \mathcal{N}_2 and furthermore $p = \text{Root}(C)$ such that $C \sim_{p'} A$ as well as $C \sim_{p''} B$ occur in \mathcal{N}_1 . In this case, we have the following indistinguishability relation between communication patterns of length 2: $A \circ A \sim_{p'} A \circ C \sim_p B \circ C \sim_{p''} B \circ B$. This argument can be applied inductively to establish the indistinguishability relation for communication patterns A^i and B^i .

LEMMA 3. *Let C_i be a connected component of $\mathcal{N}_i(\mathbf{D})$ and let A, B be communication graphs in C_i . Then A^i is connected to B^i in $I(\mathbf{D}^i)$.*

PROOF. The lemma holds immediately for $i = 1$: As a one-round communication pattern consists of only a single communication graph, $A^1 = A$ and $B^1 = B$ are both in the connected component C_1 .

Thus, we henceforth assume that $i > 1$, and prove the following claim by induction on k , for $k = 1, \dots, i$: For each edge $(A, B) \in C_i$ there is a path π_k in $I(\mathbf{D}^k)$ connecting A^k to B^k . In addition, for $k < i$, the connected component C_{i-k} of A and B in \mathcal{N}_{i-k} is such that, for every edge $e = (\sigma, \sigma') \in \pi_k$, both the round k communication graphs $\sigma(k), \sigma'(k) \in C_{i-k}$ and there is a graph $G_e \in C_{i-k}$ such that $\text{Root}(G_e) \subseteq \ell(e)$.

The base, $k = 1$, follows because $e = (A, B) \in C_i$ implies that $(A^1, B^1) \in I(\mathbf{D}^1)$, and by Corollary 1 there is $G_e \in C_{i-1}$ such that $\text{Root}(G_e) \subseteq \ell(e)$.

For the step from $k-1$ to k , $k > 1$, there exists a path $\pi_{k-1} \in I(\mathbf{D}^{k-1})$ that connects A^{k-1} to B^{k-1} . Let $e = (\sigma, \sigma') \in \pi_{k-1}$ be an arbitrary edge in π_{k-1} . By the induction hypothesis, $\sigma(k-1), \sigma'(k-1) \in C_{i-k+1}$ and there is a graph $G_e \in C_{i-k+1}$ with $\text{Root}(G_e) \subseteq \ell(e)$. Consequently, there exist paths $\tilde{\pi}_1 = (\Gamma_1, \Gamma_2, \dots, \Gamma_m)$ and $\tilde{\pi}_2 = (\Lambda_1, \Lambda_2, \dots, \Lambda_{m'})$ in C_{i-k+1} that connect $\sigma(k-1)$ to G_e and G_e to $\sigma'(k-1)$, respectively.

Consider $(\Gamma_j, \Gamma_{j+1}) \in \tilde{\pi}_1 \subseteq C_{i-k+1}$. From Corollary 1, we know that $(\Gamma_j, \Gamma_{j+1}) \in I(\mathbf{D}^1)$, which implies $\sigma \circ \Gamma_j \sim \sigma \circ \Gamma_{j+1}$. This enables us to prefix σ to each communication graph of $\tilde{\pi}_1$, which makes $\sigma \circ \tilde{\pi}_1 = (\sigma \circ \Gamma_1, \sigma \circ \Gamma_2, \dots, \sigma \circ \Gamma_m)$ a path in $I(\mathbf{D}^k)$. Following a symmetrical argument, $\sigma' \circ \tilde{\pi}_2 = (\sigma' \circ \Lambda_1, \sigma' \circ \Lambda_2, \dots, \sigma' \circ \Lambda_{m'})$ is also a path in $I(\mathbf{D}^k)$.

Moreover, since $\text{Root}(G_e) \subseteq \ell(e)$, it follows from Claim 1 that $e' = (\sigma \circ G_e, \sigma' \circ G_e) \in I(\mathbf{D}^k)$. Therefore, $\tilde{\pi}_e = (\sigma \circ \tilde{\pi}_1, e', \sigma' \circ \tilde{\pi}_2)$ is a path from $\sigma \circ \sigma(k-1)$ to $\sigma' \circ \sigma'(k-1)$ in $I(\mathbf{D}^k)$. If we substitute each edge $e \in \pi_{k-1}$ by $\tilde{\pi}_e$, we thus obtain a path π_k that connects A^k to B^k in $I(\mathbf{D}^k)$.

Now, consider any edge $e' \in \pi_k$. By construction, $e' = (\sigma \circ \Gamma_j, \sigma \circ \Gamma_{j+1})$, or $e' = (\sigma' \circ \Lambda_j, \sigma' \circ \Lambda_{j+1})$ or $e' = (\sigma \circ G_e, \sigma' \circ G_e)$. If $e' = (\sigma \circ \Gamma_j, \sigma \circ \Gamma_{j+1})$, then the round k communication graphs are Γ_j and Γ_{j+1} . Since $\tilde{\pi}_1 \in C_{i-k+1}$, it follows from Corollary 1 that $(\Gamma_j, \Gamma_{j+1}) \in C_{i-k}$, and there exists a communication graph $G_{e'} \in C_{i-k}$ with $\text{Root}(G_{e'}) \subseteq \ell((\Gamma_j, \Gamma_{j+1})) = \ell(e')$. A symmetrical argument holds for the case where $e' = (\sigma' \circ \Lambda_j, \sigma' \circ \Lambda_{j+1})$. Finally, if $e' = (\sigma \circ G_e, \sigma' \circ G_e)$, then the round k communication graphs are both G_e , which is in C_{i-k+1} by the induction hypothesis. Corollary 1 guarantees $G_e \in C_{i-k}$, and since $\text{Root}(G_e) \subseteq \ell(\sigma, \sigma')$, it follows that $\text{Root}(G_e) \subseteq \ell(\sigma \circ G_e, \sigma' \circ G_e)$. This shows that G_e is a suitable choice for $G_{e'}$, which completes the induction step. \square

THEOREM 2. *If $\mathcal{N}_i(\mathbf{D})$ contains a connected component C_i that is not root-compatible, then not all processes in all runs of a correct consensus algorithm are able to decide after i rounds under the oblivious message adversary represented by \mathbf{D} .*

PROOF. For the purpose of deriving a contradiction, suppose that the theorem does not hold. Let \mathbf{S} be a set of graphs from C_i that is not root-compatible. By Claim 2, for each $G \in \mathbf{S}$, the decision value in a run with communication pattern G^i that consists of i repetitions of G must be a value $v = x_p$ for some $p \in \text{Root}(G)$. Since \mathbf{S} is root incompatible, there exists some $H \in \mathbf{S}$ such that x_p is not a root value of H .

It follows from Lemma 3 that G^i is connected to H^i in $I(\mathbf{D}^i)$. Therefore, there is a sequence of runs $(\sigma_1 = G^i, \sigma_2, \dots, \sigma_m = H^i)$ such that σ_k is indistinguishable from σ_{k+1} . Since all processes decided $v = x_p$ in $G^i = \sigma_1$, by the validity condition of consensus, σ_2 and inductively all processes in the sequence including H^i should also decide $v = x_p$. Thus, Claim 2 yields the contradiction that H^i decided a non-broadcasted value. \square

We conclude by explaining why Theorem 2 refines the lower bound from [12, Theorem 4.10], which stated that consensus is impossible if some beta class is not root-compatible, by making the decision time explicit. In fact, in our terminology, the beta classes are the connected components of \mathcal{N}_{TD} , where TD is the smallest round such that $\mathcal{N}_{\text{TD}} = \mathcal{N}_{\text{TD}-1}$. Thus, the existence of a root-incompatible beta class is equivalent to \mathcal{N}_{TD} containing a root-incompatible connected component. Note that, since $\mathcal{N}_{\text{TD}} = \mathcal{N}_{\text{TD}-1}$, even if we remove the termination condition from Line 11 of Algorithm 1, for all $\text{TD}' \geq \text{TD} - 1$, we still have that $\mathcal{N}_{\text{TD}'} = \mathcal{N}_{\text{TD}}$, because, according to Algorithm 1, if the set of edges remains the same in an iteration of TD, then it will remain the same for all future iterations as well. Thus we can apply Theorem 2 to show that, in this case, every consensus algorithm has, for every round, a run where some process has not yet decided. As for an oblivious message adversary with a set of allowed graphs \mathbf{D} , it holds that every infinite communication pattern σ with $\sigma|_r \in \mathbf{D}^r$ for every round r satisfies $\sigma \in \mathbf{D}^\omega$ (i.e., oblivious message adversaries are limit-closed, see [33] for details), this implies that there is an infinite run where consensus is not achieved, that is, consensus is indeed impossible.

5.2 Exponential iteration complexity of the decision procedure

As we have seen above, consensus termination time is related to the iterations of the decision procedure. Informally, this is due to the fact that the information encoded in the sequence $\mathcal{N}_1, \dots, \mathcal{N}_i$ can be seen as a compact summary of

the evolution of the indistinguishability relation of the corresponding communication pattern prefixes. Thus, a lower bound on the complexity of the decision procedure immediately gives us a lower bound for the round complexity of any consensus algorithm.

In this section, we will show that the decision procedure may take an exponential number of iterations, in terms of n , until it terminates. This implies that there are oblivious message adversaries under which consensus is achievable, but reaching it takes exponential time. As already sketched at the end of Section 3, we will show this by constructing a specific instance of such a message adversary, with a set of allowed graphs $\mathbf{D} = \{G_1, \dots, G_N\}$ of size $N = 1.3^n$ (rounded down if necessary), whose indistinguishability graph $I(\mathbf{D})$ contains the following connected component:

$$G_1 \sim_{R_3} G_2 \sim_{R_4} \dots \sim_{R_{N+1}} G_N \quad (3)$$

Herein, $R_i = \text{Root}(G_i)$ for $1 \leq i \leq N$, and $R_{N+1} \neq \text{Root}(G)$ for all $G \in \mathbf{D}$. Therefore, $I(\mathbf{D})$ contains a path of length $N - 1$. Since all edges except the rightmost one are protected, Algorithm 1 only removes one edge per iteration, from right to left. More precisely, it holds that $G_1 \sim_{R_3} \dots \sim_{R_{N-i+1}} G_{N-i} \in \mathcal{N}_i$. Consequently, N iterations are needed until all edges have disappeared, which establishes our claim.

Informal overview of the definition of \mathbf{D} . First, we choose a sequence of sets $\{R_1, \dots, R_N\}$ that will play the role of root components of \mathbf{D} . We will choose those from the first half $\{p_1, \dots, p_{n/2}\}$ of the processes only. Each R_i is chosen to be unique, of the same size $n/12$, and R_i, R_{i+1} and R_{i+2} must be mutually disjoint. Note that we need N , i.e., exponentially many such R_i .

The first step in the definition of the graph G_i is to make R_i its root component, which is done by fully connecting its members to form a clique and ensuring a path to every other process. However, when doing so, we also need to guarantee that $G_i \sim_{R_{i+2}} G_{i+1}$ are the only indistinguishability relations in $I(\mathbf{D})$. We secure this by making sure that every process except for the ones in R_{i+1} and R_{i+2} can distinguish G_i from any other graph G_j , $j \neq i$. This is accomplished by adding an outgoing edge from every member of R_i to every process in $\Pi \setminus (R_{i+1} \cup R_{i+2})$, and no other outgoing edge from members of $\{p_1, \dots, p_{n/2}\}$. Since R_i is unique, any process in $\Pi \setminus (R_{i+1} \cup R_{i+2})$ will know if graph G_i is being played: This is immediately obvious for every process p in the second half $B = \{p_{n/2+1}, \dots, p_N\}$, as $\text{In}_{G_i}(p) \cap \{p_1, \dots, p_{n/2}\} = R_i$. For a process p in the “leftover set” $L_i = \Pi \setminus (B \cup R_i \cup R_{i+1} \cup R_{i+2}) \subseteq \{p_1, \dots, p_{n/2}\}$, we have $\text{In}_{G_i}(p) \cap \{p_1, \dots, p_{n/2}\} = R_i \cup \{p\}$. Since $R_i \cup \{p\}$ is larger than the size of the root components, p knows that it is not part of the root component, and can hence also uniquely determine R_i and hence the graph G_i being played. Fig. 2 illustrates this construction.

However, we must also make sure that all the members of R_{i+1} (resp. R_{i+2}) consider only G_i and G_{i-1} (resp. G_i and G_{i+1}) as possibilities for the actually played graph. This means that the in-neighborhood of any process in R_{i+1} (resp. R_{i+2}) must be the same in G_i and G_{i-1} (resp. G_i and G_{i+1}). So far, the processes in R_{i+1} or R_{i+2} do not receive any message from $\{p_1, \dots, p_n\}$, i.e., the only know that they are either in R_{i+1} or in R_{i+2} . To tell them apart, we will connect some processes in $B = \{p_{n/2+1}, \dots, p_N\}$ to the members of $R_{i+1} \cup R_{i+2}$, in a way that encodes

$i + 1$ (for the members of R_{i+1}) or $i + 2$ (for the members of R_{i+2}). A process in $R_{i+1} \cup R_{i+2}$ can hence tell from its in-neighborhood whether it belongs to R_{i+1} or R_{i+2} . More specifically, abbreviating $B[i] = \{b \in B \mid i_{b-(n/2+1)} = 1\}$,

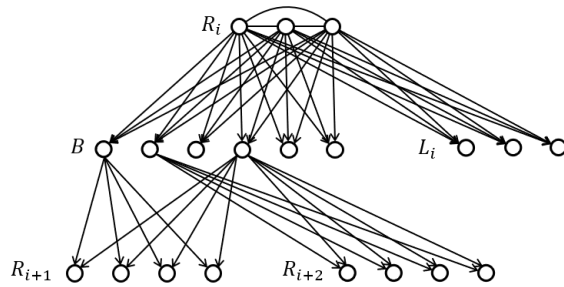


Fig. 2. A sketch of the lower bound graph G_i

where i_ℓ is the ℓ^{th} bit in the binary expansion of i , we just make sure that $\text{In}_{G_i}(p) = B[i+1]$ for every $p \in R_{i+1}$ and $\text{In}_{G_i}(p) = B[i+2]$ for every $p \in R_{i+2}$. This construction satisfies our indistinguishability requirements: Each process in R_{i+1} (resp. R_{i+2}) can tell where it belongs to, but do not know whether G_i or G_{i-1} (resp. G_i or G_{i+1}) is played.

Formal definition of the root components R_i . We define the sets R_i by splitting $\{p_1, \dots, p_{n/2}\}$ into $\{p_1, \dots, p_{n/4}\}$ and $\{p_{n/4+1}, \dots, p_{n/2}\}$, and construct the sequence R_1, R_2, \dots of root components from partitions of these ranges alternatingly: Consider all the partitions of $\{p_1, \dots, p_{n/4}\}$ into three sets of size $n/12$ each. Partition number $\ell+1$ constitutes the root components $R_{6\ell+1}, R_{6\ell+2}, R_{6\ell+3}$. Similarly, consider all the partitions of $\{p_{n/4+1}, \dots, p_{n/2}\}$ into three sets of size $n/12$ each. Set partition $\ell+1$ constitutes the root components $R_{6\ell+4}, R_{6\ell+5}, R_{6\ell+6}$.

The sequence clearly satisfies, by construction, the following properties:

- (1) $|R_i| = n/12$, since we are considering equal-sized partitions of $n/4$ processes into 3 disjoint sets.
- (2) $R_i \neq R_j$ for $i \neq j$, since all sets of the partitions are unique.
- (3) R_i, R_{i+1}, R_{i+2} are pairwise disjoint, since they are either members of the same partition and thus disjoint, or one belongs to segment $\{p_1, \dots, p_{n/4}\}$ and another to segment $\{p_{n/4+1}, \dots, p_{n/2}\}$.

The length N of the sequence is dominated asymptotically by the number of partitions of $\{p_1, \dots, p_{n/4}\}$ into three equisized sets, which is $\frac{1}{6} \binom{n/4}{n/12} \binom{n/6}{n/12}$. The definition of the binomial coefficients, along with simple bounds on the factorial function, give

$$\frac{1}{6} \binom{\frac{n}{4}}{\frac{n}{12}} \binom{\frac{n}{6}}{\frac{n}{12}} = \frac{\left(\frac{n}{4}\right)!}{6 \left(\left(\frac{n}{12}\right)!\right)^3} \geq c \frac{3^{n/4}}{n} > 1.3^n \quad (4)$$

where c is a constant and n is sufficiently large. It follows that N is exponential with respect to n .

Formal definition of G_i . We are now ready to define the graphs G_i , recall also Fig. 2. Let $B = \{p_{n/2+1}, \dots, p_n\}$. For each $1 \leq i \leq N$, the graph G_i is composed of disjoint 5 node sets: B, R_i, R_{i+1}, R_{i+2} , where $R_i, R_{i+1}, R_{i+2} \subseteq \{p_1, \dots, p_{n/2}\}$, $B = \{p_{n/2+1}, \dots, p_n\}$, and $L_i = \Pi \setminus (B \cup R_i \cup R_{i+1} \cup R_{i+2})$.

Connect every two nodes in R_i by bi-directional edges, forming a clique. From each node in R_i , add a directed edge to each node in $B \cup L_i$. Finally, for an index i , let $B[i] = \{b \in B \mid i_{b-(n/2+1)} = 1\}$, where i_ℓ is the ℓ^{th} bit in the binary expansion of i . Add an edge from each node of $B[i]$ to each node of R_{i+1} , and similarly, from each node of $B[i+1]$ to each node of R_{i+2} .

We are now ready to show that the so-constructed graphs form an indistinguishability chain according to Eq. (3).

CLAIM 5. For $1 \leq i \leq N$, we have $B[i] \neq \emptyset$, and for $1 \leq i < j \leq N$, we have $B[i] \neq B[j]$.

PROOF. As $N = 1.3^n$, we find $\log_2(N) < n/2$, so each 1-bit of i is represented by a process in B , which ends up being in $B[i]$. This establishes the second assertion. The first one is now trivial, as $i \geq 1$. \square

CLAIM 6. For $1 \leq i \leq N$, we have $\text{Root}(G_i) = R_i$.

PROOF. This is immediate from the graph's definition. In G_i , all nodes in R_i are connected to one another and have no incoming edges from any node not in R_i . From each of them, there is a direct edge to all nodes of $B \cup L_i$. Moreover, by Claim 5, there is at least one process $b \in B[i]$, so there is a path from each node in R_i , through b , to each node in $R_{i+1} \cup R_{i+2}$. \square

CLAIM 7. We have $G_i \sim_{R_{i+2}} G_{i+1}$ for $1 \leq i \leq N-1$, and these are the only indistinguishability relations in the graph.

PROOF. As we have already explained in the informal overview, in G_i , every process that is not in $R_{i+1} \cup R_{i+2}$ can determine that the graph is G_i from its in-neighborhood. This is immediately obvious for processes in B , and also possible for a process $p \in L_i$ by observing $|\text{In}_{G_i}(p)| = n/12 + 1$ and removing itself from it for determining R_i .

For a process $p \in R_{i+1}$ (resp. R_{i+2}), it holds by construction that $\text{In}_{G_i}(p) = B[i + 1] = \text{In}_{G_{i-1}}(p)$ (resp. $\text{In}_{G_{i-1}}(p) = B[i + 2] = \text{In}_{G_{i+1}}(p)$), and that G_{i-1} (resp. G_{i+1}) is the only other graph besides G_i where the in-neighborhood of p is the same. \square

Our lower bound is now easy to prove.

THEOREM 3. *There is an oblivious message adversary under which consensus is solvable, but for which the decision procedure takes exponential time to terminate.*

PROOF. Let $\mathbf{D} = \{G_i \mid 1 \leq i \leq N\}$, where $N = 1.3^n$ for n begin sufficiently large for Eq. (4) to hold. We consider Algorithm 1, and show, by induction on the iteration number i , that after iteration i the graphs G_1, \dots, G_{N-i+1} constitute the only nontrivial connected component in \mathcal{N}_i .

The base case is $\mathcal{N}_1 = I(\mathbf{D})$, where the graphs G_1, \dots, G_N are connected by Claim 7. For the inductive step $i - 1 \rightarrow i$, $i > 1$, assume G_1, \dots, G_{N-i+2} is the only nontrivial connected component in \mathcal{N}_{i-1} , and consider iteration i .

For G_1, \dots, G_{N-i+1} , every two consecutive graphs G_j, G_{j+1} with $1 \leq j \leq N - i$ are indistinguishable for a set R_{j+2} by Claim 7, which is the root component of G_{j+2} by Claim 6. Since G_{j+2} is in the same connected component as G_j and G_{j+1} in \mathcal{N}_{i-1} , the edge $G_j \sim_{R_{j+2}} G_{j+1}$ is incorporated by the algorithm in \mathcal{N}_i .

On the other hand, the edge $G_{N-i+1} \sim_{R_{N-i+3}} G_{N-i+2}$ of \mathcal{N}_{i-1} is not added to \mathcal{N}_i . This is since R_{N-i+3} is the root component of G_{N-i+3} , which is not in the nontrivial connected component of \mathcal{N}_i . Since all the root components have equal sizes and are distinct, R_{N-i+3} cannot be contained in any other root component either. This completes the induction step.

It follows that the algorithm takes $N = 1.3^n$ iterations to complete. Upon completion, each connected component of \mathcal{N}_N is a single, root-compatible graph, so consensus is solvable under \mathbf{D} . \square

5.3 Exponential termination time of consensus

From Theorem 2, we immediately obtain a termination time lower bound of $\Omega(\text{TD})$ for solving consensus. Consequently, the message adversary used in (the proof of) Theorem 3, where $\text{TD} = N = 1.3^n$ for sufficiently large n , reveals a lower bound that is exponential in n .

We will now adapt the message adversary from Theorem 3 in Section 5.2 to show that the termination time of consensus may actually be $\Omega(n1.3^n)$. More specifically, in the graph G_i shown in Fig. 2, we replace the direct edges from R_i to B by a path consisting of processes taken from a set $P \subseteq \{p_{n/2+1}, \dots, p_n\}$ with $|P| = \Omega(n)$ (i.e., taken away from the original B), as illustrated in Fig. 3.

In more detail, we change the graph construction from Section 5.2 as follows:

- $B = \{n/2 + 1, \dots, 0.9n\}$ and $P = \{0.9n + 1, \dots, n\}$;
- Add the directed edges $(p, p + 1)$ for all $p \in P \setminus \{n\}$;
- Instead of an edge from each node of R_i to each node of B , add an edge from each node of R_i to $h = 0.9n + 1$, and from n to each node of B .

Let $h = 0.9n$ be the first node on the inserted path. Whereas our new construction introduced the additional indistinguishability $G_i \sim_p G_j$ for all $p \in (B \cup P) \setminus \{h\}$ for any $G_i, G_j \in \mathbf{D}$, it does not affect the iteration complexity of

the decision procedure, since no $R \subseteq (B \cup P)$ ever occurs as a root component in a graph of \mathbf{D} . Thus, all edges e with $\ell(e) \subseteq (B \cup P)$ are removed in the first iteration of the decision procedure, according to Corollary 1.

It is easy to see that Claim 5 still holds, as we have $\log_2(N) < 0.4n$, and Claim 6 holds by construction. Regarding Claim 7, the original indistinguishability relations still hold, but are now expanded by additional indistinguishabilities labeled by a process $p \in (P \cup B) \setminus \{h\}$, which are removed in the first iteration of the decision procedure.

The crucial property of our new construction is that any G_i, G_{i+1} , when repeated for $0.1n$ rounds, yield indistinguishable communication patterns.

CLAIM 8. $G_i^r \sim_p G_{i+1}^r$ for all $r \leq 0.1n$ and all $p \in R_{i+2}$.

PROOF. Observe that, by construction, we have $\text{In}_{G_i}(p) = \text{In}_{G_{i+1}}(p)$ for all $p \in X = P \setminus \{h\} \cup B \cup R_{i+2}$. The claim follows, because every path from a process outside X to a process in R_{i+2} has length at least $|P| + 1$. It thus takes at least $|P| + 1$ repetitions of G_i , respectively G_{i+1} , until a process of $\Pi \setminus X$ reached a process of R_{i+2} . Since $|P| = 0.1n$, in a round $r \leq 0.1n$, the nodes of R_{i+2} have hence the same view in both G_i^r and G_{i+1}^r . \square

The following Lemma 4 shows that we can even “inflate” arbitrary communication patterns of the message adversary from Section 5.2:

LEMMA 4. Consider $(\sigma, \sigma') \in I(\mathbf{D}^k)$, where \mathbf{D} is the oblivious message adversary of Section 5.2. Let $\tilde{\mathbf{D}}$ be the modified message adversary of Section 5.3, and $\tilde{\sigma}$ resp. $\tilde{\sigma}'$ in $\tilde{\mathbf{D}}^{(k, 0.1n)}$ be the communication pattern obtained from replacing every round i graph $\sigma(i)$ resp. $\sigma'(i)$ according to Fig. 2 by $0.1n$ instances of the corresponding graph according to Fig. 3. Then, $(\tilde{\sigma}, \tilde{\sigma}') \in I(\tilde{\mathbf{D}}^{0.1nk})$.

PROOF. We prove, by induction over $k \geq 1$, that (i) the $0.1nk$ prefixes $\tilde{\sigma}|_{0.1nk}$ and $\tilde{\sigma}'|_{0.1nk}$ satisfy $\tilde{\sigma}|_{0.1nk} \sim_R \tilde{\sigma}'|_{0.1nk}$ for $R = \ell(\sigma, \sigma') \neq \emptyset$, and (ii) that $\tilde{\sigma}|_{0.1nk} \sim_B \tilde{\sigma}'|_{0.1nk}$ if and only if $\sigma|_k \sim_B \sigma'|_k$ for the processes $B = \{p_{n/2+1}, \dots, p_n\}$. Note carefully that $\sigma \sim_R \sigma'$ also implies $\sigma|_k \sim_R \sigma'|_k$, as well as $\sigma(k) \sim_R \sigma'(k)$. As a consequence, there is some i such that, for every k , either $\sigma(k) = G_i$ and $\sigma'(k) = G_{i+1}$ (or vice versa), with $R = R_{i+2}$, or else $\sigma(k) = \sigma'(k)$.

For the induction basis $k = 1$, the only non-trivial case is $\sigma|_1 = \sigma(1) = G_i \in \mathbf{D}$ and $\sigma'|_1 = \sigma'(1) = G_{i+1} \in \mathbf{D}$, and $R = R_{i+2}$. From Claim 8, we get $\tilde{\sigma}|_{0.1n} \sim_R \tilde{\sigma}'|_{0.1n}$ as needed for (i). As for (ii), the length $0.1n$ of the path P in Fig. 3 ensures that all processes in B have the same distinguishing power in both the original and in the inflated prefix.

For the induction step $k - 1 \rightarrow k$, $k > 1$, we assume for our hypothesis that $\tilde{\sigma}|_{0.1n(k-1)} \sim_R \tilde{\sigma}'|_{0.1n(k-1)}$ and that all processes in B have the same distinguishing power. Assume for a contradiction for (i) that $\tilde{\sigma}|_{0.1nk} \not\sim_R \tilde{\sigma}'|_{0.1nk}$, i.e., some process $p \in R$ can distinguish the two prefixes. Consider the round k graphs $\sigma(k)$ and $\sigma'(k)$. If $\sigma(k) = \sigma'(k) = G_j \in \mathbf{D}$, we immediately get a contradiction, since appending $0.1n$ instances $\hat{G}_j^{0.1n}$ of the corresponding $\hat{G}_j \in \tilde{\mathbf{D}}$ to both $\tilde{\sigma}|_{0.1n(k-1)}$ and $\tilde{\sigma}'|_{0.1n(k-1)}$ cannot break their indistinguishability for p .

So let us assume w.l.o.g. $G_i = \sigma(k)$ and $G_{i+1} = \sigma'(k)$ with $R = R_{i+2}$. Since we know from Claim 8 that the corresponding graphs in $\tilde{\mathbf{D}}$ ensure $\hat{G}_i^{0.1n} \sim_p \hat{G}_{i+1}^{0.1n}$, the information that allows p to distinguish $\tilde{\sigma}|_{0.1nk}$ and $\tilde{\sigma}'|_{0.1nk}$ was relayed to it from some informed process q' during the last $0.1n$ rounds. Since R_{i+2} only has incoming edges from B in

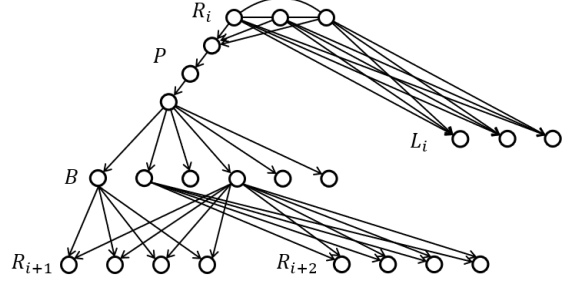


Fig. 3. A sketch of the extended lower bound graph G_i

Fig. 3, there exists an informed process $q \in B$ that relayed this information to p by the last of these rounds. This q must have been informed at the latest in round $0.1nk - 1$. Since the path P in Fig. 3 has length $0.1n$, however, R_i (resp. R_{i+1}) cannot be the source of information that allows q to distinguish $\tilde{\sigma}|_{0.1nk}$ and $\tilde{\sigma}'|_{0.1nk}$. Consequently, q must already have had information to distinguish $\tilde{\sigma}|_{0.1n(k-1)}$ and $\tilde{\sigma}'|_{0.1n(k-1)}$. From (ii) of our induction hypothesis, we can infer that this is also true in the original $\sigma|_{k-1}$ and $\sigma'|_{k-1}$. Since q sends a message to R_{i+2} in round k here, this would contradict $\sigma|_k \sim_R \sigma'|_k$, and therefore completes the induction step for (i).

The induction step for (ii) is trivial, as the processes in B only get information from the respective root component, either directly (in the original prefix) or delayed via the path P (in the inflated one). The induction hypothesis hence immediately carries over from $k - 1$ to k . \square

Lemma 4 immediately gives us the consensus termination time for our new message adversary:

THEOREM 4. *There is an oblivious message adversary for which solving consensus takes $\Omega(n1.3^n)$ rounds.*

PROOF. Consider any two indistinguishable communication patterns σ, σ' of the message adversary of Theorem 3 on the path between G_1^{N-1} and G_2^{N-1} in $I(\mathbf{D})^{N-1}$. As $\text{TD} = N = 1.3^n$, Lemma 3 guarantees that this path exists. Lemma 4 immediately provides us with inflated communication patterns $\mu, \mu' \in I(\mathbf{D})^{0.1n(N-1)}$ for our new message adversary, which are also indistinguishable. Together, they form a path between $G_1^{0.1n(N-1)}$ and $G_2^{0.1n(N-1)}$ in $I(\mathbf{D})^{0.1n(N-1)}$. Since the root components $R_1 = \text{Root}(G_1)$ and $R_2 = \text{Root}(G_2)$ are disjoint, not all processes can have decided by round $0.1n(N - 1)$, as claimed. \square

6 THE SOURCE OF CONSENSUS TIME COMPLEXITY

In this section, we want to investigate whether the number of iterations TD of the decision procedure is the sole cause for a large time complexity of consensus in an oblivious message adversary. Before we do so, however, let us briefly reiterate what we have achieved so far. In Theorem 1 we have seen that consensus can be solved after $c(n - 1) \cdot \text{TD}$ rounds, whereas Theorem 4 revealed that there are in fact oblivious message adversaries where consensus takes up to n TD rounds to terminate and TD may be exponential in n . Thus in these cases a time complexity exponential in n is asymptotically tight for solving consensus under an oblivious message adversary. As we know that the consensus time complexity is always at most $c(n - 1) \cdot \text{TD}$, and since we have examples where it is at least n TD, it might hence be tempting to assume that TD also determines the termination time of consensus in all cases. In this section, we will see that this is not the case, as, to the contrary, there are instances where the decision procedure terminates after a constant number of iterations while the consensus time complexity is exponential in n . We now proceed to show how to derive such an instance.

6.1 A partition of an oblivious message adversary

Before going into the details of how to construct a message adversary with the desired property of incurring a large time complexity of consensus while maintaining a low TD, we investigate an abstract property that, if satisfied by an oblivious message adversary \mathbf{D} for a parameter t , leads to a consensus time complexity in the order of t . Informally, this property is that there exists a partition S_1, \dots, S_t of \mathbf{D} such that S_1 is connected in the indistinguishability graph $I(\mathbf{D})$ and all the edges that make up this connection are protected by the communication graphs of S_2 . Similarly, S_2 is connected in $I(\mathbf{D})$ and all of the edges in this connection, along with the ones from S_1 , are protected by the communication graphs of S_3 and so on. Our claim is that if some t round communication patterns exist that have no common broadcaster

and whose round $1 \leq r \leq t$ communication graphs are picked from S_r , then consensus is impossible by round t . The reason for this, as shown in more detail below, is that the set of communication patterns $S_1 \circ \dots \circ S_t$ is connected in the indistinguishability graph $I(\mathbf{D}^t)$, because each S_r can maintain the connectivity of $S_1 \circ \dots \circ S_{r-1}$ in $I(\mathbf{D}^{r-1})$ as all the edges relevant for this connectivity are protected by the communication graphs of S_r .

Formally, we express this property as follows:

DEFINITION 2. *Let S_1, \dots, S_t be a partition of \mathbf{D} with the following properties, for $1 \leq i \leq t$:*

- (i) *Each S_i is connected. That is, for each G, G' in S_i , there is a path from G to G' in the indistinguishability graph $I(\mathbf{D})$ that consists only of elements from S_i .*
- (ii) *The edges of the subgraph of $I(\mathbf{D})$, induced by $\bigcup_{j=1}^{i-1} S_j$, are protected by the communication graphs of S_i .*
- (iii) *There is no process p such that every communication pattern of $\Sigma = S_1 \circ \dots \circ S_t$ is broadcastable by p .*

Given this partition, we show in Claim 9 below that Σ is connected in $I(\mathbf{D}^t)$, which shows that consensus is impossible after t rounds: If all processes do decide after t rounds in all runs with a communication pattern of Σ , they all decide the same value because Σ is connected in $I(\mathbf{D}^t)$. Thus, in some run with communication pattern $\sigma \in \Sigma$, the decision is on an input of a process p even though σ is not broadcastable by p , which contradicts Claim 2.

CLAIM 9. *The communication patterns of $\Sigma_t = S_1 \circ \dots \circ S_t$ are pairwise connected to each other in $I(\mathbf{D}^t)$.*

PROOF. Let $I(\mathbf{D})[S]$ denote the subgraph of $I(\mathbf{D})$, induced by the set of communication graphs S . We show an even stronger claim, namely that there is a set of edges E_t that connects Σ_t in $I(\mathbf{D}^t)$ such that for each $e \in E_t$ there is an $e' \in I(\mathbf{D})[\bigcup_{j \leq t} S_j]$ with the same label $\ell(e) = \ell(e')$. We show this by induction on k with $\Sigma_k = S_1 \circ \dots \circ S_k$.

The base of the induction $k = 1$ follows directly from property (i) of Definition 2, as S_1 is connected in $I(\mathbf{D})$.

For the step from k to $k + 1$, the induction hypothesis is that there are edges E_k that connect Σ_k such that for every $e \in E_k$ there is an $e' \in I(\mathbf{D})[\bigcup_{j \leq k} S_j]$ with $\ell(e) = \ell(e')$. We use the graphs of S_{k+1} to extend Σ_k to Σ_{k+1} while maintaining the connectivity of Σ_{k+1} as follows.

For every $\sigma_1, \sigma_2 \in \Sigma_k$ with $e = (\sigma_1, \sigma_2) \in E_k$, we add to Σ_{k+1} the extensions $\sigma_1 \circ G$ and $\sigma_2 \circ G$ such that $G \in S_{k+1}$ and G protects e . Such a communication graph G exists because of property (ii) of Definition 2 and because there is an edge $e' \in I(\mathbf{D})[\bigcup_{j \leq k} S_j]$ with $\ell(e) = \ell(e')$ by hypothesis.

Finally, for all extensions $\sigma_1 \circ G, \sigma_2 \circ G$ and $\sigma_2 \circ G', \sigma_3 \circ G'$ added to Σ_{k+1} in this way, by property (i) of Definition 2, there is a path π from G to G' in $I(\mathbf{D})$ that consists only of graphs $G'' \in S_{k+1}$. We can thus add all the communication patterns $\{\sigma_2 \circ G'' : G'' \in \pi\}$ to Σ_{k+1} as well: This maintains the connectivity of Σ_{k+1} and ensures the induction hypothesis as the path π lies entirely in $I(\mathbf{D})[S_{k+1}]$ by property (i) of Definition 2. \square

6.2 An example: choosing the processes

We now construct a set of communication graphs that can be partitioned in accordance with Definition 2, into $t = 1.07^n$ sets. For a set Π of n processes, let $m = \lceil \frac{n}{10} \rceil$. We construct a message adversary with a partition on it, $\mathbf{D} = \bigcup_{i=1}^t S_i$, where each S_i is a set of $2i + 1$ graphs, denoted $S_i = \{G_{i,j} \mid 1 \leq j \leq 2i + 1\}$. Each graph $G_{i,j}$ is defined by a partition of the process set Π as

$$\Pi = \begin{cases} B \cup R_j \cup U_i \cup U'_i \cup L_{i,j} & \text{for } j = 1, \\ B \cup R_j \cup U_i \cup L_{i,j} & \text{for } j \text{ even,} \\ B \cup R_j \cup U'_i \cup L_{i,j} & \text{for } j \geq 3 \text{ odd.} \end{cases}$$

The process sets are $B = [5m + 1, n]$, which is fixed for all i, j . R_j with $|R_j| = m$, which constitutes the root component of all graphs $G_{i,j}$. U_i, U'_i , with $|U_i| = |U'_i| = m$. The set $L_{i,j}$ is defined to be the set of all the remaining processes. We choose processes for these sets by induction on i , as follows. For the base, we show how to construct the sets for the communication graphs of $S_1 = \{G_{1,1}, G_{1,2}, G_{1,3}\}$.

- (b1) $R_1 = [4m + 1, 5m]$
- (b2) $R_2 \subseteq [1, 2m]$, $|R_2| = m$, chosen arbitrarily
- (b3) $R_3 \subseteq [2m + 1, 4m]$, $|R_3| = m$, chosen arbitrarily
- (b4) $U_1 \subseteq [2m + 1, 4m]$ different from R_3
- (b5) $U'_1 \subseteq [1, 2m]$ different from R_2

We proceed with the inductive step of our construction. For this we assume that we are given R_1, \dots, R_{2i+1} and U_i, U'_i , and show how to construct R_{2i+2}, R_{2i+3} and U_{i+1}, U'_{i+1} .

- (s1) We let $R_{2i+2} = U'_i$
- (s2) We let $R_{2i+3} = U_i$
- (s3) We let U_{i+1} be an arbitrary subset of $[2m + 1, 4m]$ of size m , different from $R_2, R_4 \dots R_{2i+2}$
- (s4) We let U'_{i+1} be an arbitrary subset of $[1, 2m]$ of size m , different from $R_3, R_5 \dots R_{2i+3}$

Note that steps (s1) and (s2) are always possible, as long as the sets U_i and U'_i are defined. To see that we can repeat step (s3) for t times, note that there are $\binom{2m}{m}$ many ways to choose a set $U_{i+1} \subseteq [2m + 1, 4m]$ of size m . We have

$$\binom{2m}{m} \geq \frac{(2m)^m}{m^m} = 2^m \geq 2^{n/10} > 1.07^n$$

and the claim follows. The claim for (s4) is analogous.

6.3 An example: the graph structure

We now show how to combine the sets R_j, U_i, U'_i, B , and $L_{i,j}$ in $G_{i,j}$ to obtain an oblivious message adversary that has a partition as described in Definition 2 for $t = 1.07^n$ (for an illustration, see Fig. 4). While the choice processes of R_j is independent of i , the edges between them in $G_{i,j}$ will depend crucially on i .

The graph $G_{i,j}$ always contains a directed cycle in R_j in increasing order of the process identifiers. Note that $|R_j| = m$ and each process already has one incoming edge from the preceding process, and thus there are $m - 2$ other potential incoming edges we can choose to add. Hence, there are $m \cdot 2^{m-2} > t$ (for n large enough) possible interconnects for R_j , and for each i we choose a different one.

We define the other edges of $G_{i,j}$ as follows. Each graph contains edges from all process of R_j to all processes of B and $L_{i,j}$. For an index i , let $B[i] = \{b \in B \mid i_{b-(5m+1)} = 1\}$, where i_h is the h^{th} bit in the binary expansion of i . Note that for $i \neq i', 1 \leq i, i' \leq t$ we have $B[i] \neq B[i']$, i.e. all the bits of i are represented in $B[i]$, since $\log_2 t < 0.1n$.

The rest of the edges depend on j , as follows.

- For $j = 1$, add an edge from each node of $B[i]$ to each node of $U_i \cup U'_i \cup L_{i,j}$.
- For j even, add an edge from each node of $B[i]$ to each node of $U_i \cup L_{i,j}$.
- For $j \geq 3$ odd, add an edge from each node of $B[i]$ to each node of $U'_i \cup L_{i,j}$.

6.4 An example: properties of the adversary

Finally, let us establish our main claim, namely that the above construction indeed yields an oblivious message adversary where the consensus time complexity $t = 1.07^n$ grows exponentially with n , yet $\text{TD} = 2$, a constant. In the remainder of this section, we show these properties for the oblivious message adversary \mathbf{D} constructed above. First, we show that \mathbf{D} partitions as described in Definition 2.

CLAIM 10. *The sets S_1, \dots, S_t are a partition according to Definition 2.*

PROOF. For property (i), the connectivity of S_i , pick any $G_{i,j} \in S_i$. We show this graph is indistinguishable to some processes from $G_{i,1}$, and thus the graph is connected by an edge to $G_{i,1}$ in $I(\mathbf{D})$. If j is odd, the in-neighborhood of every process of U'_i is the same in $G_{i,j}$ and in $G_{i,1}$, namely $B[i]$. Similarly, if j is even, every process of U_i has $B[i]$ as its in-neighborhood in $G_{i,j}$, and this is also the case for $G_{i,1}$.

To prove property (ii), which states that the communication graphs of S_i protect the edges that were used to connect S_1, \dots, S_{i-1} , it suffices to show that for every $1 \leq i' < i$, there are communication graphs $G, G' \in S_i$ such that $\text{Root}(G) \subseteq U_{i'}$ and $\text{Root}(G') \subseteq U'_{i'}$. For a given $1 \leq i' < i$, note that the graphs $G_{i,2i'+2}, G_{i,2i'+3} \in S_i$ satisfy $R_{2i'+2} = U_{i'}$ and $R_{2i'+3} = U'_{i'}$ by construction.

For property (iii), which states that there is no process by which all communication patterns of $\Sigma = S_1 \circ \dots \circ S_t$ are broadcastable, let us investigate the processes that were able to broadcast in $(G_{i,2})_{i=1}^t \in \Sigma$ and $(G_{i,3})_{i=1}^t \in \Sigma$. We observe that, by (b2) and (b3), for all $1 \leq i \leq t$, $\text{Root}(G_{i,2}) = R_2 \subseteq [1, 2m]$ and $\text{Root}(G_{i,3}) = R_3 \subseteq [2m+1, 4m]$ and thus $R_2 \cap R_3 = \emptyset$. As the broadcasters of $(G_{i,2})_{i=1}^t$ are R_2 and the broadcasters of $(G_{i,3})_{i=1}^t$ are R_3 , property (iii) holds. \square

CLAIM 11. *The decision procedure terminates after $\text{TD} = 2$ iterations on \mathbf{D} .*

PROOF. First, note that all the roots R_j are contained in $[1, 5m]$, while $B = [m5+1, n]$, hence no edge of $I(\mathbf{D})$ labeled by only processes of B will be preserved after the first iteration. Similarly, we can ignore processes of B in the labels, when considering the preservation of the edges.

We show that in the first iteration of the decision procedure, none of the edges of $I(\mathbf{D})$ that connect graphs from different sets in the partition $\mathbf{D} = \bigcup_{i=1}^t S_i$ are preserved. Consider $G_{i,j} \in S_i, G_{i',j'} \in S_{i'}, i \neq i'$, such that $G_{i,j} \sim_\ell G_{i',j'}$. Note that in $G_{i,j}$, the processes of U_i (or U'_i if j is odd) and $L_{i,j}$ have $B[i]$ as their incoming edges, while the corresponding processes in $G_{i',j'}$ have $B[i']$, and $B[i] \neq B[i']$, so none of U_i (or U'_i), $U_{i'}$ (or $U'_{i'}$), $L_{i,j}$ and $L_{i',j'}$ intersect ℓ .

Hence, the only processes in ℓ that can occur in a root component of a graph of \mathbf{D} are processes of R_j and $R_{j'}$. Let us study $|R_j \cap R_{j'} \cap \ell|$: if $j \neq j'$ then $R_j \neq R_{j'}$ so $|R_j \cap R_{j'} \cap \ell| < |R_j| = m$; if $j = j'$, then the fact that the choice of interconnects for R_j in $G_{i,j}$ depends on i guarantees that at least one process of R_j is not in $R_j \cap R_{j'} \cap \ell$, and again $|R_j \cap R_{j'} \cap \ell| < m$. As any root component $R_{j''}$ of a graph in \mathbf{D} has $|R_{j''}| = m$, no such root component satisfies $R_{j''} \subseteq R_j \cap R_{j'} \cap \ell$, and the edge ℓ is not being preserved in the first iteration.

Second, we show that in the second iteration of the decision procedure, none of the edges in $I(\mathbf{D})$ that is within a set S_i is preserved. Assume for contradiction that for some i , there are graphs $G_{i,j}, G_{i,j'}, G_{i,j''} \in S_i, j \neq j'$ such that $G_{i,j} \sim_\ell G_{i,j'}$ and $R_{j''} \subseteq \ell$. All the processes of $B, L_{i,j}$ and $L_{i,j'}$ have incoming edges from R_j (or $R_{j'}$), and since $R_j \neq R_{j'}$ none of these processes appear in ℓ . Note that $1 \leq j'' \leq 2i+1$, and the sets U_i and U'_i are chosen to be different from R_1, \dots, R_{2i+1} , which implies $R_{j''} \neq U_i, U'_i$.

If $j = 1$, only processes of $U_i \cup U'_i$ can appear in ℓ . This is because in $G_{i,1}$, processes of R_1 do not have any incoming edge from B , which they have in all other graphs of S_i , and processes of $L_{i,1}$ have incoming edges from R_1 , which no process has in any other graph of S_i . Therefore $R_{j''} \subseteq \ell \subseteq U_i \cup U'_i$, where $U_i \subseteq [2m+1, 4m]$ and $U'_i \subseteq [1, 2m]$. But either $R_{j''} \subseteq [1, 2m]$ or $R_{j''} \subseteq [2m+1, 4m]$, and $|R_{j''}| = |U_i| = |U'_i|$, so either $R_{j''} = U_i$ or $R_{j''} = U'_i$, a contradiction.

If j is even, $R_{j''} \subseteq \ell \subseteq R_j \cup U_i$, as any process not in $R_j \cup U_i$ has incoming edges from all processes of R_j in $G_{i,j}$, which it does not have in $G_{i,j'}$. We have $R_j \subseteq [1, 2m]$ (as j is even) and $U_i \subseteq [2m+1, 4m]$, while either $R_{j''} \subseteq [1, 2m]$ or $R_{j''} \subseteq [2m+1, 4m]$. So, either $R_{j''} = R_j$ or $R_{j''} = U_i$. This can only occur if $j = j''$: the sets R_k are different for different indices k , and U_i is chosen to be different from R_1, \dots, R_{2i+1} . The case of $j > 1$ odd is analogous, and we conclude that $j = j''$ in both cases. The same analysis applies for j' , and so we have $j = j'' = j'$, a contradiction. \square

From this, we conclude the main theorem of this section.

THEOREM 5. *There exists an oblivious message adversary with exponential consensus time complexity in spite of a constant iteration complexity TD of the decision procedure.*

7 CONCLUSIONS

This paper presented a simple procedure for deciding whether solving consensus is possible under a given oblivious message adversary. Whereas it can be viewed as an early terminating version of the abstract beta class characterization by Couloma, Godard, and Peters [12], our formulation turned out to be instrumental for characterizing the, to the

best of our knowledge, previously unknown termination time of distributed consensus under a given message adversary. We discovered a close relation between the number of iterations of the decision algorithm and the consensus termination time, and the importance of the existence and number of root-compatible connected components in the refined indistinguishability graph.

Our work opens several interesting avenues for future work. For example, while we have presented a combinatorial approach, it would be interesting to study the time complexity of the consensus problem from a topological perspective as well. It would further be interesting to fully understand the implications of our approach on distributed information dissemination problems such as broadcast, and explore alternative adversarial models. We also plan to conduct an empirical study of our algorithms to complement the theoretical perspective and analysis presented in this paper.

ACKNOWLEDGMENTS

Research supported by the Austrian Science Fund (FWF) project DELTA (Dependable Network Data Plane for the Cloud), I 5025-N, a joint project with Hungarian National Research, Development and Innovation Office NKFIH (co-PI: Gabor Retvari).

REFERENCES

- [1] Ittai Abraham, Dahlia Malkhi, et al. The blockchain consensus layer and bft. *Bulletin of EATCS*, 3(123), 2017.
- [2] Yehuda Afek and Eli Gafni. Asynchrony from synchrony. In *Distributed Computing and Networking*, volume 7730 of *Lecture Notes in Computer Science*, pages 225–239. Springer Berlin Heidelberg, 2013.
- [3] Hagit Attiya and Armando Castañeda. A non-topological proof for the impossibility of k-set agreement. *Theor. Comput. Sci.*, 512:41–48, 2013.
- [4] Hagit Attiya, Armando Castañeda, Maurice Herlihy, and Ami Paz. Bounds on the step and namespace complexity of renaming. *SIAM J. Comput.*, 48(1):1–32, 2019.
- [5] Martin Biely, Peter Robinson, and Ulrich Schmid. Agreement in directed dynamic networks. In *Proceedings 19th International Colloquium on Structural Information and Communication Complexity (SIROCCO'12)*, LNCS 7355, pages 73–84. Springer-Verlag, 2012.
- [6] Martin Biely, Peter Robinson, Ulrich Schmid, Manfred Schwarz, and Kyrill Winkler. Gracefully degrading consensus and k-set agreement in directed dynamic networks. *Theoretical Computer Science*, 726:41–77, 2018.

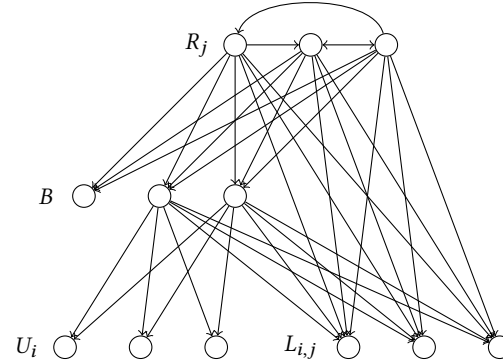


Fig. 4. Topology of $G_{i,j}$ for even j , used to establish an exponential consensus time complexity in spite of a constant TD.

- [7] Martin Biely, Ulrich Schmid, and Bettina Weiss. Synchronous consensus under hybrid process and link failures. *Theoretical Computer Science*, 412(40):5602 – 5630, 2011. <http://dx.doi.org/10.1016/j.tcs.2010.09.032>.
- [8] Ofer Biran, Shlomo Moran, and Shmuel Zaks. A combinatorial characterization of the distributed 1-solvable tasks. *Journal of algorithms*, 11(3):420–440, 1990.
- [9] Armando Castañeda, Pierre Fraigniaud, Ami Paz, Sergio Rajsbaum, Matthieu Roy, and Corentin Travers. A topological perspective on distributed network algorithms. In *Structural Information and Communication Complexity - 26th International Colloquium, SIROCCO*, pages 3–18, 2019.
- [10] Bernadette Charron-Bost, Matthias Függer, and Thomas Nowak. Approximate consensus in highly dynamic networks: The role of averaging algorithms. In *Automata, Languages, and Programming*, volume 9135 of *LNCS*, pages 528–539. Springer Berlin Heidelberg, 2015.
- [11] Bernadette Charron-Bost and André Schiper. The Heard-Of model: computing in distributed systems with benign faults. *Distributed Computing*, 22(1):49–71, April 2009.
- [12] Étienne Coulouma, Emmanuel Godard, and Joseph G. Peters. A characterization of oblivious message adversaries for which consensus is solvable. *Theor. Comput. Sci.*, 584:80–90, 2015.
- [13] Tristan Fevat and Emmanuel Godard. Minimal obstructions for the coordinated attack problem and beyond. In *25th IEEE International Symposium on Parallel and Distributed Processing, IPDPS*, pages 1001–1011, 2011.
- [14] Michael J. Fischer, Nancy A. Lynch, and M. S. Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM*, 32(2):374–382, April 1985.
- [15] Matthias Függer, Thomas Nowak, and Manfred Schwarz. Tight bounds for asymptotic and approximate consensus. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing, PODC '18*, pages 325–334, New York, NY, USA, 2018. ACM.
- [16] Matthias Függer, Thomas Nowak, and Kyrill Winkler. On the radius of nonsplit graphs and information dissemination in dynamic networks. *Discrete Applied Mathematics*, 282:257–264, 2020.
- [17] Eli Gafni. Round-by-round fault detectors: unifying synchrony and asynchrony. In *Proceedings of the Seventeenth Annual ACM Symposium on Principles of Distributed Computing*, pages 143–152. ACM Press, 1998.
- [18] Maurice Herlihy, Dmitry N. Kozlov, and Sergio Rajsbaum. *Distributed Computing Through Combinatorial Topology*. Morgan Kaufmann, 2013.
- [19] Idit Keidar and Alex Shraer. Timeliness, failure detectors, and consensus performance. In *Proceedings of the twenty-fifth annual ACM SIGACT-SIGOPS symposium on Principles of Distributed Computing (PODC'06)*, pages 169–178, New York, NY, USA, 2006. ACM Press.
- [20] Dmitry N. Kozlov. Structure theory of flip graphs with applications to weak symmetry breaking. *CoRR*, abs/1511.00457, 2015.
- [21] Dmitry N. Kozlov. *Combinatorial Topology of the Standard Chromatic Subdivision and Weak Symmetry Breaking for Six Processes*, pages 155–194. Springer International Publishing, Cham, 2016.
- [22] F. Kuhn and R. Oshman. Dynamic networks: Models and algorithms. *SIGACT News*, 42(1):82–96, 2011.
- [23] Fabian Kuhn, Nancy A. Lynch, and Rotem Oshman. Distributed computation in dynamic networks. In *STOC*, pages 513–522, 2010.
- [24] Fabian Kuhn, Rotem Oshman, and Yoram Moses. Coordinated consensus in dynamic networks. In *Proceedings of the 30th annual ACM SIGACT-SIGOPS symposium on Principles of distributed computing, PODC '11*. ACM, 2011.
- [25] Calvin Newport, David Kotz, Yougu Yuan, Robert S. Gray, Jason Liu, and Chip Elliott. Experimental Evaluation of Wireless Simulation Assumptions. *SIMULATION: Transactions of The Society for Modeling and Simulation International*, 83(9):643–661, September 2007.
- [26] Thomas Nowak, Ulrich Schmid, and Kyrill Winkler. Topological characterization of consensus under general message adversaries. In *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing, PODC*, pages 218–227, 2019. (full version: <http://arxiv.org/abs/1905.09590>).
- [27] Diego Ongaro and John Ousterhout. In search of an understandable consensus algorithm. In *Proc. USENIX Annual Technical Conference (ATC)*, pages 305–319, 2014.
- [28] Nicola Santoro and Peter Widmayer. Time is not a healer. In *Proc. 6th Annual Symposium on Theor. Aspects of Computer Science (STACS'89)*, LNCS 349, pages 304–313. Paderborn, Germany, February 1989. Springer-Verlag.
- [29] Nicola Santoro and Peter Widmayer. Agreement in synchronous networks with ubiquitous faults. *Theoretical Computer Science*, 384(2–3):232–249, October 2007.
- [30] Ulrich Schmid, Bettina Weiss, and Idit Keidar. Impossibility results and lower bounds for consensus under link failures. *SIAM Journal on Computing*, 38(5):1912–1951, 2009.
- [31] Manfred Schwarz, Kyrill Winkler, and Ulrich Schmid. Fast consensus under eventually stabilizing message adversaries. In *Proceedings of the 17th International Conference on Distributed Computing and Networking, ICDCN '16*, pages 7:1–7:10, New York, NY, USA, 2016. ACM.
- [32] Kyrill Winkler and Ulrich Schmid. An overview of recent results for consensus in directed dynamic networks. *Bulletin of the EATCS*, 128, 2019.
- [33] Kyrill Winkler, Ulrich Schmid, and Yoram Moses. A characterization of consensus solvability for closed message adversaries. In *23rd International Conference on Principles of Distributed Systems OPODIS*, volume 153 of *LIPICs*, pages 17:1–17:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.
- [34] Kyrill Winkler, Manfred Schwarz, and Ulrich Schmid. Consensus in directed dynamic networks with short-lived stability. *Distributed Computing*, 32(5):443–458, 2019.
- [35] Martin Zeiner, Manfred Schwarz, and Ulrich Schmid. On linear-time data dissemination in dynamic rooted trees. *Discrete Applied Mathematics*, 255:307 – 319, 2019.