The Time Complexity of Consensus Under Oblivious Message Adversaries

Kyrill Winkler
ITK Engineering, Wien, Austria

Ami Paz
LISN – CNRS & Paris-Saclay University, France

Hugo Rincon Galeana
TU Wien, Austria

Stefan Schmid
TU Berlin, Gemrany
Fraunhofer SIT, Darmstadt, Germany

Ulrich Schmid
TU Wien, Austria

Abstract

We study the problem of solving consensus in synchronous directed dynamic networks, in which communication is controlled by an oblivious message adversary that picks the communication graph to be used in a round from a fixed set of graphs $\mathbf{D}$ arbitrarily. In this fundamental model, determining consensus solvability and designing efficient consensus algorithms is surprisingly difficult. Enabled by a decision procedure that is derived from a well-established previous consensus solvability characterization for a given set $\mathbf{D}$, we study, for the first time, the time complexity of solving consensus in this model: We provide both upper and lower bounds for this time complexity, and also relate it to the number of iterations required by the decision procedure. Among other results, we find that reaching consensus under an oblivious message adversary can take exponentially longer than both deciding consensus solvability and broadcasting the input value of some unknown process to all other processes.

2012 ACM Subject Classification Theory of computation → Distributed algorithms; Networks

Keywords and phrases dynamic networks, oblivious message adversaries, consensus, time complexity

Digital Object Identifier 10.4230/LIPIcs.ITCS.2023.100

Funding This research has received funding from the German Federal Ministry of Education and Research (BMBF), 6G-RIC, grant 16KISK020K; Vienna Science and Technology Fund (WWTF) project ICT19-045; Austrian Science Fund (FWF) projects ADynNet P28182, ByzDEL P33600, and netIDEE SCIENCE P33775-N.
Kyrill Winkler: Vienna Science and Technology Fund (WWTF) project ICT19-045, 2020-2024.
Hugo Rincon Galeana: Supported by the Doctoral College on Resilient Embedded Systems at TU Wien.

1 Introduction

Consensus, the 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 deterministic consensus has been studied intensively for several decades, it is still unknown whether and, in particular, how quickly it can be achieved in fundamental models such as the one we study here: synchronous dynamic directed networks controlled by an oblivious message adversary.

14th Innovations in Theoretical Computer Science Conference (ITCS 2023).
Editor: Yael Tauman Kalai; Article No. 100; pp. 100:1–100:28
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The study of such dynamic 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 are often *asymmetric* (e.g., in optical or in wireless networks) [27]. It is also of fundamental theoretical interest, as solving consensus in dynamic directed networks is known to be significantly more difficult [33, 32] than solving consensus in dynamic networks with bidirectional links [26].

Focusing on the worst-case perspective, we consider 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 indicate which process can successfully send a message to which other process in that round. We specifically consider the fundamental oblivious message adversary model [12] introduced by Coulouma, Godard and Peters, which is specified via a set $\mathbf{D}$ of allowed communication graphs from which the message 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 may vary greatly from one round to the next (e.g., if a message failed on a link in some round, messages may still be sent on this link in successive rounds). 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 [30], 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 would be possible for a smaller set $\mathbf{D}$. However, to the best of our knowledge, nothing is known about the complexity of *deciding* consensus solvability given $\mathbf{D}$, i.e., using a centralized algorithm and, in particular, about the time complexity of *distributed consensus* in the cases where it is possible.

Contributions. In this paper, we present both a practical decision procedure for consensus solvability, based on the beta classes introduced in [12], and provide the first study of the consensus time complexity in this model. In more detail:

- We provide a centralized decision procedure for determining consensus solvability for a given oblivious message adversary $\mathbf{D}$ and bound its complexity. Compared to the beta class criterion proposed in [12], our decision procedure has the advantage of being explicit, simple and early deciding.
- We provide both upper and lower bounds for the worst-case time complexity of distributed consensus[1] for oblivious message adversaries. Interestingly, this time complexity may depend not only on the number of processes n and the number of iterations TD required by our centralized decision procedure, but sometimes also on the number of connected components in the outcome of the decision procedure after TD iterations. Indeed, by

[1] We emphasize that *worst-case time complexity of distributed consensus* means that no correct consensus algorithm can terminate faster than the lower bound in all possible runs, and that there is some consensus algorithm that terminates within the upper bound in all runs. When we talk about the time complexity of consensus in our paper, we always mean worst-case time complexity.

means of two deliberately crafted examples of oblivious message adversaries, we show that consensus termination times exponential in n can happen whether or not TD is exponential in n. Note that the dependency on the number of connected components could not be inferred from the existing beta class results [12].

Our results also shed an interesting new light on the relationship between distributed consensus and broadcasting: As the input value of some process is known to reach all other processes in linear time under any oblivious message adversary [13, 14, 37, 18], one might be tempted to expect that consensus can also be achieved quickly. Our results show that, quite to the contrary, reaching consensus can take exponential time, even in cases where the decision procedure terminates in only a few iterations.

Paper organization. The remainder of this paper is organized as follows. After a brief survey of related work in Section 2, we introduce our formal model and terminology in Section 3. The description and analysis of our decision procedure and our consensus algorithm are presented in Section 4 and Section 5, respectively, and our lower bound results are presented in Section 6. A fully worked out scenario showing constant iteration complexity vs. exponential consensus termination time is provided in Section 7. We conclude our contribution and discuss directions for future work in Section 8.

2 Related Work

Consensus is a fundamental task in distributed computing, and the question whether and when consensus is possible has engaged researchers at least since the influential impossibility result by Fischer, Lynch, and Paterson [16] 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 [29, 24, 9, 34, 1].

Research on deterministic consensus in synchronous message-passing systems subject to link failures dates back to the seminal paper by Santoro and Widmayer [30], 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 [31, 32, 11, 7, 12, 10, 17]. For example, in [32], Schmid, Weiss and Keidar 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 [19, 21, 11], like the heard-of model by Charron-Bost and Schiper [11], and different agreement problems like approximate and asymptotic consensus have been studied in these models [10, 17]. In many of these and similar works on consensus [15, 5, 33, 6, 36, 28, 9], a model is considered in which, in each round, a digraph is picked from a set of allowed 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 [34].

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 [25, 26]. 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. These systems can be roughly described using one specific oblivious message adversary, containing all transitively closed tournaments. Wait-free computation in this context is often studied using topological tools [20, 3, 4, 22, 23]. 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 [31]. 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 so-called beta equivalence classes of this relation. The decision procedure from Section 4 can be seen as an explicit computation of these beta classes. Moreover, [12] also presents a distributed consensus algorithm that, essentially, computes the beta classes. However, in stark contrast to our paper, neither the iteration complexity of our decision procedure nor the time complexity of distributed consensus has been studied in this or any other paper we are aware of.

3 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 *choice 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 chosen by an oblivious message adversary. A message from p to q is delivered in round r only if the communication graph of round r contains the edge (p, q). Since every process knows its own current view, we just assume that the communication graph always contains all the self-loops. We use $\mathrm{In}_G(v)$ to 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 communication graphs, which (along with the initial views of all processes and the choice function Δ) will uniquely define a run of the system. In the oblivious message adversary model, there is a set $\mathbf{D}$ of allowed graphs, and the admissible communication patterns are all infinite 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 $\mathbf{D}^\omega$.

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 $i \leq r$ 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 $\mathrm{Root}(G)$ 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 $p \in \mathrm{Root}(G)$. 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., $\bigcap_{G \in \mathbf{S}} \mathrm{Root}(G) \neq \emptyset$. 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 $\mathrm{view}_\sigma(p, 0) = \{(p, 0, x_p)\}$ and, for $r > 0$, $\mathrm{view}_\sigma(p, r) = (p, r, V_\sigma(p, r-1))$, where $V_\sigma(p, r-1) = \{\mathrm{view}_\sigma(q, r-1) : (q, p) \in \sigma(r)\}$.

For notational simplicity, we will subsequently use the tuple (p, r) 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 $\mathrm{view}_\sigma(p_i, r' + i - 1) \in V_\sigma(p_{i+1}, r' + i - 1)$ 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* r in case of infinite patterns. Formally, $\sigma \sim_p \sigma' \Leftrightarrow \mathrm{view}_\sigma(p, r) = \mathrm{view}_{\sigma'}(p, r)$ if σ and σ' are r-round patterns, and $\sigma \sim_p \sigma' \Leftrightarrow \mathrm{view}_\sigma(p, r) = \mathrm{view}_{\sigma'}(p, r)$ 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 $\ell : E(I(\Sigma)) \to 2^\Pi$ 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 $\mathrm{Root}(G) \subseteq \ell((\sigma, \sigma'))$. Then, the communication patterns $\sigma \circ G$ and $\sigma' \circ G$ are also indistinguishable for the processes in $\mathrm{Root}(G)$: in G, these processes only receive messages from other members of $\mathrm{Root}(G)$, 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 $\mathrm{Root}(G) \subseteq \ell(e)$ then the edge $e' = (\sigma \circ G, \sigma' \circ G)$ is in $I(\mathbf{D}^{r+1})$ and its label $\ell(e')$ satisfies $\mathrm{Root}(G) \subseteq \ell(e') \subseteq \ell(e)$.

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 $p \in \mathrm{Root}(G)$. By the definition of a root component, we have $\mathrm{In}_G(p) \subseteq \mathrm{Root}(p)$, so each process q with $(q, r) \in \mathrm{view}_{\sigma \circ G}(p, r + 1)$, is in $\mathrm{Root}(G)$, and satisfies $\mathrm{view}_\sigma(q, r) = \mathrm{view}_{\sigma'}(q, r)$, because $\mathrm{Root}(G) \subseteq \ell(e)$. 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, $\text{view}_\sigma(q, r) \in V_\sigma(q, r)$. ◁

In the ***consensus problem***, each process p has an input value $x_p \in V$, taken from some (usually 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 consider only full information protocols, our consensus algorithm is actually a collection of choice functions. For every $p \in \Pi$, the choice 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^r_σ 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) corresponding to σ is the sequence $(C^r_\sigma)_{r\geq 0}$. 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 [35, 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 γ 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 γ. By validity, $P \neq \emptyset$. Let γ_j denote the run that is the same as γ, 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 γ_j for $0 \leq j \leq k$. Thus in the run γ_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 $\gamma \sim \gamma_0 = \varepsilon$.

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 γ_j is identical to γ_{j+1} except for the input of $p_{i_{j+1}}$, we have $\gamma_j \sim_q \gamma_{j+1}$. As all processes decide by round r in γ_j, and because they decide v by hypothesis, q and, by agreement, all processes decide v in γ_{j+1}. ◁

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. ◁

4 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 $\mathcal{N}_1 = I(\mathbf{D}), \mathcal{N}_2, \ldots$ of refinements of $I(\mathbf{D})$, and use the last graph $\mathcal{N}_{\mathrm{TD}}$ to decide if consensus is solvable under the oblivious message adversary $\mathbf{D}$. Here, TD is the number of iterations of the decision procedure. Our decision procedure can be viewed as an iterative computation of the beta classes introduced in [12], which is also early-terminating: it may terminate early if broadcastability of the intermediate result is already guaranteed. As an additional feature, it reveals a certain 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 $A, B \in \mathbf{D}$ are connected by an edge if there is a process p that has the same set of incoming edges in A and in B. The algorithm then computes a sequence $(\mathcal{N}_i)_{i \geq 1}$ 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 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 $\mathrm{Root}(G) \subseteq \ell(e)$ (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 12 and 14, the root-compatibility of the connected components of the refined indistinguishability graph is precisely what is required to make consensus solvable.

Whereas the number of iterations of our decision procedure could be exponential (see Claim 7), every iteration can be performed efficiently, as its main components require merely the computation of reachability in graphs: In each $G \in \mathbf{D}$ during the initialization, and in $\mathcal{N}_i$ in iteration i. The running time is hence polynomial in n and $|\mathbf{D}|$ for both the initialization and for each iteration:

▷ Claim 4. The initialization phase of Algorithm 1 can be implemented in $O(|\mathbf{D}|^2 n^3)$ time, and each of its iterations can be implemented in $O(|\mathbf{D}|^3 n^2)$ time.

Proof. Recall that the transitive closure of a (directed or undirected) graph $G = (V, E)$ is a graph $H = (V, \hat{E})$ such that $(u, v) \in \hat{E}$ if there is a (directed) path from u to v in G. The transitive closure can be computed in $O(|V|^3)$ time using combinatorial algorithms (e.g., the Floyd–Warshall algorithm), and slightly faster using fast matrix multiplication.

For the initialization, compute for each $G \in \mathbf{D}$ its transitive closure, and set $\mathrm{Root}(G)$ as the set of nodes that can reach all other nodes – we have seen that being in $\mathrm{Root}(G)$ and being able to reach all other nodes are equivalent conditions. Overall, this takes $O(|\mathbf{D}| n^3)$ time. Then, build the graph $I(\mathbf{D})$ and assign labels to its edges by going over all pairs $(A, B) \in \mathbf{D} \times \mathbf{D}$ and checking for every $p \in \Pi$ whether $\mathrm{In}_A(p) = \mathrm{In}_B(p)$, which takes $O(|\mathbf{D}|^2 n^2)$ time.

For iteration i, start by computing the transitive closure of $\mathcal{N}_{i-1}$ (in fact, this is done at the end of the previous iteration). For each edge $(A, B) \in E_{i-1}$, go over all graphs G reachable from A to see if one of them satisfies $\mathrm{Root}(G) \subseteq \ell((A, B))$. This takes $O(|\mathbf{D}|^3 n^2)$

time overall. To check the stopping condition, compare E_i and E_{i-1} (taking $O(|\mathbf{D}|^2)$ time), compute the transitive closure of $\mathcal{N}_i$ to find its connected components (taking $O(|\mathbf{D}|^3)$ time), and finally, for each of the at most $|\mathbf{D}|$ connected components, go over Π and mark for each process $p \in \Pi$ if there is a graph in the connected component (which has size at most $|\mathbf{D}|$) that does not have p in its root. Overall, this takes no more than $O(|\mathbf{D}|^3 n^2)$ time. ◁

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 A, B, we have $\ell((A, B)) = \{p \in \Pi : A \sim_p B\} = \{p \in \Pi : \mathrm{In}_A(p) = \mathrm{In}_B(p)\}$.

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

```
   Input: A set of allowed graphs $\mathbf{D}$
   Output: The refined indistinguishability graph $\mathcal{N}_{\mathrm{TD}}$. Consensus is solvable if and only if
           all connected components of $\mathcal{N}_{\mathrm{TD}}$ are root-compatible.
   // Initialization:
 1 $i \leftarrow 1$
 2 $\mathcal{N}_1 \leftarrow I(\Sigma)$ for $\Sigma = \mathbf{D}$
   // 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 the endpoints of $e$ in $\mathcal{N}_{i-1}$
 8         if $\exists G \in \mathbf{G} : \mathrm{Root}(G) \subseteq \ell(e)$ 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 $\mathcal{N}_{i-1}$
```

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 5.** *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 graph G_e with $\mathrm{Root}(G_e) \subseteq \ell(e)$, 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 $\mathrm{Root}(G_e) \subseteq \ell(e)$ connected to A and B in $\mathcal{N}_{i-1}$. ◀

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 6.** *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 $\mathrm{Root}(G_e) \subseteq \ell(e)$.*

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 7. The number of iterations of the decision procedure, TD, satisfies $\mathrm{TD} < |\mathbf{D}|$ and $\mathrm{TD} < 2^n$.

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

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

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 $\mathbf{C}_i(\mathbf{G})$ 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, $|\mathbf{C}_i(\mathbf{G})| < |\mathbf{C}_{i-1}(\mathbf{G})|$; 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. ◁

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. As we already know that $\mathrm{TD} < |\mathbf{D}|$, one may wonder how large that gap between them can be. The following examples show an exponential gap, with $\mathrm{TD} = 1$ and $|\mathbf{D}|$ exponential in n. We first present such a scenario in which consensus is solvable: 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 each connected component in $I(\mathbf{D})$ consists of a single communication graph. An example where the decision procedure terminates quickly 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 $\mathrm{Root}(T) \subseteq \ell(e)$.

On the other hand, the question arises whether there are examples where TD is (almost) the same as $|\mathbf{D}|$. This is of course the case if $|\mathbf{D}|$ is small, but we answer this question affirmatively also for the case where TD is even exponential in n, by constructiong an explicit example (Section 6). In a nutshell, we choose a set of communication graphs $\mathbf{D} = \{G_1, \dots, G_{\mathrm{TD}}\}$, 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 $|\mathrm{Root}(G)| = |\mathrm{Root}(G')|$, but if $G \neq G'$ then $\mathrm{Root}(G) \neq \mathrm{Root}(G')$. Furthermore, we let

$$G_1 \sim_{R_3} G_2 \sim_{R_4} G_3 \sim_{R_5} \dots \sim_{R_{\mathrm{TD}}} G_{\mathrm{TD}-1} \sim_S G_{\mathrm{TD}}, \tag{1}$$

where $R_i = \mathrm{Root}(G_i)$ and S is a nonempty set such that no $G \in \mathbf{D}$ satisfies $\mathrm{Root}(G) \subseteq S$. Here, the decision procedure can remove only the rightmost edge $\sim_S$ in the first iteration, only the edge $\sim_{R_{\mathrm{TD}}}$ 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 6 described above), or it might be impossible, as in the instance

$$G_1' \sim_{R_3'} G_2' \sim_{R_1'} G_3' = G_1 \sim_{R_3} G_2 \sim_{R_4} \ldots \sim_{R_{\mathrm{TD}}} G_{\mathrm{TD}-1} \sim_S G_{\mathrm{TD}}$$

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 it is TD that actually determines the worst-case termination time of distributed consensus. Interestingly, this is not the case. Complementing the result of Theorem 12 established in Section 5, we show in Section 7 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 at least exponential in n. This example also reveals that the time complexity of distributed consensus is not solely determined by the need to compute (overapproximations of) the beta classes of [12].

5 Time Complexity of Consensus

In this section, we study the worst-case time complexity of consensus, and also ascertain our claim from Section 4, 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 $\mathcal{N}_{\mathrm{TD}}$ are root-compatible.

For solving consensus, we use the fact that non-connectivity in $\mathcal{N}_{\mathrm{TD}}$ implies non-connectivity in $I(\mathbf{D}^{(n-1)\,\mathrm{TD}+1})$, in the following sense: Let $\mathbf{C}_1$ and $\mathbf{C}_2$ be two different connected components of $\mathcal{N}_{\mathrm{TD}}$, and $t > (n-1)\,\mathrm{TD}$. Then, any two communication patterns $\sigma_1 \in \mathbf{C}_1^t$ and $\sigma_2 \in \mathbf{C}_2^t$, consisting only of graphs of $\mathbf{C}_1$ and $\mathbf{C}_2$, 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 $\mathcal{N}_{\mathrm{TD}}$. Note that this choice guarantees that graphs from at least one connected component are used at least t times. From here, a choice function Δ_p can be easily defined by (i) for each connected component $\mathcal{C}$ of $I(\mathbf{D}^{ct})$, choosing one of its broadcasters, denoted $b(\mathcal{C})$, and (ii) if p's view is consistent with a graph sequence σ, and σ belongs to a connected component $\mathcal{C}$ of $I(\mathbf{D}^{ct})$, then p decides on the input $x_{b(\mathcal{C})}$ of $b(\mathcal{C})$, for which $\mathrm{view}_\sigma(b(\mathcal{C}), 0, x_{b(\mathcal{C})})$ must already be present in p's view.

It is rather immediate that such a procedure solves consensus, given the mapping $b(\mathcal{C})$ which we define in the remainder of this section: Termination follows from the existence of the mapping $b(\mathcal{C})$; 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 $\mathcal{C}$ 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 $\pi = (\sigma_0, \ldots, \sigma_s)$ 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 $0 \le i < s$. Given such a path and $r' \le r$, we write $\pi|_{r'}$ to denote the path $(\sigma_0|_{r'}, \ldots, \sigma_\ell|_{r'})$ 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 8 shows that edges, and hence paths, between communication patterns in $I(\mathbf{D}^r)$ are preserved when omitting some round r'.

▶ **Corollary 8.** *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. ◀

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

▶ **Corollary 9.** *Let $\mathbf{D}$ be a set of allowed graphs and $0 < r' < r$ integers. Consider an edge $e = (\sigma, \sigma') \in I(\mathbf{D}^r)$ such that $e' = (\sigma|_{r'}, \sigma'|_{r'}) \in I(\mathbf{D}^{r'})$ 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

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

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 $\overline{U}_j = \Pi \setminus U_j$ and consider the cut $(U_j, \overline{U}_j)$ 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 $\overline{U}_j \neq \emptyset$, then there is a node $p'' \in \overline{U}_j$, and a path in $\sigma(r_j)$ from p' to p''; this path must cross an edge $\tilde{e}_j$ from U_j to $\overline{U}_j$.

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 $\overline{U}_j$. As, for every $q \in \Pi \setminus \ell(e')$ in the definition if U_j in Equation (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 $I(\mathbf{D}^r)$. ◀

We proceed with Lemma 10, which generalizes and formalizes chains like Equation (1), made up of connected subgraphs $\mathcal{S}_1, \dots, \mathcal{S}_i$ which are interconnected in a chain. It uses protected edges in order to delay the separation of root-incompatible connected components as much as possible, namely, by removing the interconnects between S_j and S_{j+1} in $\mathcal{N}_{i-j}$, i.e., from right (i) to left (1).

▶ **Lemma 10.** *Given an oblivious 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 6, $\text{Root}(G) \subseteq \ell(e)$. 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 $e \in \mathcal{N}_{i'+1}$. ◀

We are now ready for 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 two sets $\mathbf{C}_1$ and $\mathbf{C}_2$, respectively, is related to the connectivity of $\mathbf{C}_1$ and $\mathbf{C}_2$ in $\mathcal{N}_{\text{TD}}$ as computed by Algorithm 1.

▶ **Lemma 11.** *Given an oblivious message adversary $\mathbf{D}$, let $\mathbf{C}$ constitute a connected component of $\mathcal{N}_{\text{TD}}$ 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 $\mathcal{N}_{\text{TD}}$, contradicting the fact that $\mathbf{C}$ is a connected component of $\mathcal{N}_{\text{TD}}$. 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 $1 \leq j < i = TD$, 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 10, 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 8, 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 $1 \leq m \leq \mathrm{TD}$, property (1) together with $r_1^* = 1$ guarantees $r_m^* \leq (n-1)(m-1) + 1 \leq (n-1) \cdot \mathrm{TD} = r$, 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 9, 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 $\mathrm{Root}(G) \subseteq \ell(e)$. 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 $\bigcup_{k=1}^{m+1} \pi_k$. ◀

We can now state the main theorem of this section, an upper bound on the time complexity of consensus.

▶ **Theorem 12.** *Let $\mathbf{D}$ be the set of allowed communication graphs of an oblivious message adversary. If the connected components of $\mathcal{N}_{\mathrm{TD}}(\mathbf{D})$ are root-compatible, then consensus is solvable by round $c(n-1)(\mathrm{TD}+1)$, where c is the number of connected components in $\mathcal{N}_{\mathrm{TD}}$.*

Proof. We show that every connected component of the indistinguishability graph $I(\mathbf{D}^t)$ is broadcastable for $t = c(n-1)(\mathrm{TD}+1)$. This implies the theorem, because there exists a mapping for every connected component $\mathcal{C}$ of $I(\mathbf{D}^t)$ to a process p, such that p is a broadcaster in every communication pattern of $\mathcal{C}$. More specifically, as $\mathcal{C}$ is an indistinguishability component, there is, for every process q and every $\sigma \in \mathbf{D}^t$, a map $\mathrm{view}_\sigma(q,t) \mapsto p$ 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 $\mathcal{C}$, every process has thus already learned the input x_p of p, which is a valid decision value. Our 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 $\sigma(i)$, $i = 1 \dots, c(n-1)(\mathrm{TD}+1)$ appearing in it. By the pigeon-hole principle, at least one connected component $\mathbf{C}$ of $\mathcal{N}_{\mathrm{TD}}$ must supply $(n-1)(\mathrm{TD}+1)$ of these graphs, when counted with repetitions. That is, there is a set $R \subseteq \{1, \dots, c(n-1)(\mathrm{TD}+1)\}$, with $|R| = (n-1)(\mathrm{TD}+1)$, such that every r_i with $i \in R$ satisfies $\sigma(r_i) \in \mathbf{C}$. Note that the occurrence of $n-1$ or more graphs from $\mathbf{C}$ in σ already suffices to ensure that it is broadcastable by every process $p \in \bigcap_{G \in \mathbf{C}} \mathrm{Root}(G)$, i.e., that every process $q \in \Pi$ has $(p, 0, x_p) \in \mathrm{view}_\sigma(q,t)$.

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 $\sigma'(r_i) \in \mathbf{C}$, σ' would also be broadcastable by $\bigcap_{G \in \mathbf{C}} \mathrm{Root}(G)$, so assume that this is not the case. There are hence at most $n-2$ indices $r_j \in R$ where $\sigma'(r_j) \in \mathbf{C}$. Let $R' \subseteq R$ with $|R'| = (n-1) \cdot \mathrm{TD}$ 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 $((n-1)\,\mathrm{TD})$-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 8, ρ and ρ' are connected in $I(\mathbf{D}^{(n-1)\,\mathrm{TD}})$. This, however, contradicts Lemma 11, because $\rho \in \mathbf{C}^{(n-1)\,\mathrm{TD}}$ and $\rho' \in \bar{\mathbf{C}}^{(n-1)\,\mathrm{TD}} \subseteq \bar{\mathbf{C}} \times \mathbf{D}^{(n-1)\,\mathrm{TD}-1}$ by construction, where $\mathbf{C}$ is a connected component in $\mathcal{N}_{\mathrm{TD}}$ and $\bar{\mathbf{C}}$ is its complement. ◀

The result of Theorem 12 suggests that, besides the termination time TD of the decision procedure (which can be attributed to the complexity of finding broadcastable components) and the number of processes $n-1$ (which accounts for the worst-case information propagation time from root components to all other processes), the number of connected components c in $\mathcal{N}_{\mathrm{TD}}$ might also cause exponential time complexity for solving distributed consensus. And indeed, the scenarios presented in Section 7, where TD is constant, prove this to be true.

6 Lower Bounds

This section complements our positive results above by studying lower bounds. In the following, we first establish a relationship between the iteration complexity of the decision procedure and the termination time of consensus. We 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.

6.1 Decision complexity and consensus termination time

First, we present a relationship (Theorem 14) 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 $\mathrm{Root}(G)$, 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 in Lemma 13. 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 $I(\mathbf{D}^i)$. The proof 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 (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 5. 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 13.** *Let $\mathcal{C}_i$ be a connected component of $\mathcal{N}_i(\mathbf{D})$ and let A, B be communication graphs in $\mathcal{C}_i$. 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 $\mathcal{C}_1$.

Thus, we henceforth assume that $i > 1$, and prove the following claim by induction on k, for $k = 1, \dots, i$: For each edge $(A, B) \in \mathcal{C}_i$ 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 $\mathcal{C}_{i-k}$ 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 $\sigma(k), \sigma'(k) \in \mathcal{C}_{i-k}$ and there is a graph $G_e \in \mathcal{C}_{i-k}$ such that $\text{Root}(G_e) \subseteq \ell(e)$.

The base, $k = 1$, follows because $e = (A, B) \in \mathcal{C}_i$ implies that $(A^1, B^1) \in I(\mathbf{D}^1)$, and by Corollary 5 there is $G_e \in \mathcal{C}_{i-1}$ 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, $\sigma(k-1)$, $\sigma'(k-1) \in \mathcal{C}_{i-k+1}$ and there is a graph $G_e \in \mathcal{C}_{i-k+1}$ 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 $\mathcal{C}_{i-k+1}$ that connect $\sigma(k-1)$ to G_e and G_e to $\sigma'(k-1)$, respectively.

Consider $(\Gamma_j, \Gamma_{j+1}) \in \tilde{\pi}_1 \subseteq \mathcal{C}_{i-k+1}$. From Corollary 5, 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 $\tilde{\pi}_1 \in \mathcal{C}_{i-k+1}$, it follows from Corollary 5 that $(\Gamma_j, \Gamma_{j+1}) \in \mathcal{C}_{i-k}$, and there exists a communication graph $G_{e'} \in \mathcal{C}_{i-k}$ 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 $\mathcal{C}_{i-k+1}$ by the induction hypothesis. Corollary 5 guarantees $G_e \in \mathcal{C}_{i-k}$, 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. ◀

▶ **Theorem 14.** *If $\mathcal{N}_i(\mathbf{D})$ contains a connected component $\mathcal{C}_i$ 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 $\mathcal{C}_i$ 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 13 that G^i is connected to H^i in $I(\mathbf{D}^i)$. Therefore, there is a sequence of runs $(\sigma_1 = G^i, \sigma_2, \ldots, \sigma_m = H^i)$ 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. ◀

We conclude by explaining why Theorem 14 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 round number i and hence a time complexity lower bound explicit. In fact, in our terminology, the beta classes are the connected components of $\mathcal{N}_{\text{TD}}$, 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 $\mathcal{N}_{\text{TD}}$ 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 14 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 [35] for details), this implies that there is an infinite run where consensus is not achieved.

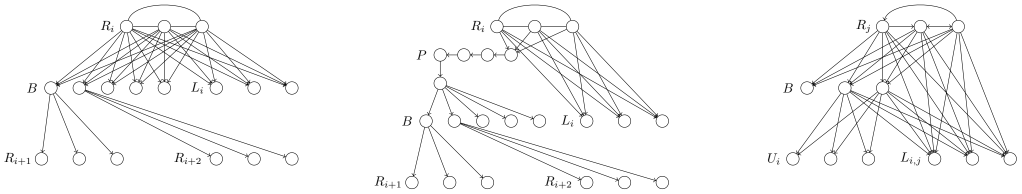
6.2 Exponential iteration complexity of the decision procedure

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 $\mathcal{N}_1, \ldots, \mathcal{N}_i$ 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 iteration complexity of the decision procedure immediately gives us a lower bound for the round complexity of any consensus algorithm.

In this section, we show that the decision procedure may take an exponential (in n) number of iterations 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 4, we show this by constructing a specific instance of such an oblivious message adversary, with a set of $N = 1.3^n$ allowed graphs $\mathbf{D} = \{G_1, \ldots, G_N\}$, whose indistinguishability graph $I(\mathbf{D})$ contains the following connected component:

$$G_1 \sim_{R_3} G_2 \sim_{R_4} G_3 \sim_{R_5} \ldots \sim_{R_{N+1}} G_N \tag{3}$$

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 $G_1 \sim_{R_3} \ldots \sim_{R_{N-i+1}} G_{N-i} \in \mathcal{N}_i$. Consequently, N iterations are needed until all edges have disappeared, which establishes our claim.



(a) The lower bound graph G_i.

(b) G_i for extended lower bound.

(c) $G_{i,j}$ for even j, exponential complexity with $\mathrm{TD} = 2$.

Figure 1 Lower bound communication graphs.

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 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 $\mathrm{In}_{G_i}(p) \cap \{p_1, \dots, p_{n/2}\} = R_i$. 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 $\mathrm{In}_{G_i}(p) \cap \{p_1, \dots, p_{n/2}\} = R_i \cup \{p\}$. 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. Figure 1a 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 $B[i] = \{b \in B \mid i_{b-(n/2+1)} = 1\}$, where i_ℓ is the ℓ^{th} bit in the binary expansion of i, we just make sure that $\mathrm{In}_{G_i}(p) = B[i+1]$ for every $p \in R_{i+1}$ and $\mathrm{In}_{G_i}(p) = B[i+2]$ 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 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 consider all the partitions of $\{p_{n/4+1}, \ldots, p_{n/2}\}$ 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 $\{p_1, \ldots, p_{n/4}\}$ and another to segment $\{p_{n/4+1}, \ldots, p_{n/2}\}$.

The length N of the sequence is dominated asymptotically by the number of partitions of $\{p_1, \ldots, p_{n/4}\}$ 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

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

where c is a constant and n is sufficiently large. It follows that N is exponential in n.

Formal definition of G_i. We are now ready to define the graphs G_i, recall also Figure 1a. Let $B = \{p_{n/2}+1, \ldots, p_n\}$. 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 $R_i, R_{i+1}, R_{i+2} \subseteq \{p_1, \ldots, p_{n/2}\}$, $B = \{p_{n/2}+1, \ldots, p_n\}$, 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 Equation (3).

▷ Claim 15. 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 $i \geq 1$. ◁

▷ Claim 16. For $1 \leq i \leq N$, we have $\mathrm{Root}(G_i) = R_i$.

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 15, 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 $R_{i+1} \cup R_{i+2}$. ◁

▷ Claim 17. 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 $|\operatorname{In}_{G_i}(p)| = n/12 + 1$ 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 $\operatorname{In}_{G_i}(p) = B[i+1] = \operatorname{In}_{G_{i-1}}(p)$ (resp. $\operatorname{In}_{G_{i-1}}(p) = B[i+2] = \operatorname{In}_{G_{i+1}}(p)$), 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. ◁

Our lower bound is now easy to prove.

▶ **Theorem 18.** *There is an oblivious message adversary under which consensus is solvable, but for which the decision procedure takes time exponential in n 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 Equation (4) to hold. We consider Algorithm 1, and show, by induction on the iteration number i, that after iteration i the graphs $G_1, \ldots, G_{N-i+1}$ constitute the only nontrivial connected component in $\mathcal{N}_i$.

The base case is $\mathcal{N}_1 = I(\mathbf{D})$, where the graphs $G_1, \ldots, G_N$ are connected by Claim 17. For the inductive step $i-1 \to i$, $i > 1$, assume $G_1, \ldots, G_{N-i+2}$ is the only nontrivial connected component in $\mathcal{N}_{i-1}$, and consider iteration i.

For $G_1, \ldots, G_{N-i+1}$, 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 17, which is the root component of G_{j+2} by Claim 16. 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 $\mathbf{D}$. ◀

6.3 Exponential termination time of consensus

From Theorem 14, we immediately obtain a termination time lower bound of $\Omega(\mathrm{TD})$ for solving consensus. Consequently, the oblivious message adversary used in (the proof of) Theorem 18, where $\mathrm{TD} = N = 1.3^n$ for sufficiently large n, reveals a lower bound that is exponential in n.

We now adapt the oblivious message adversary from Theorem 18 (Section 6.2) to get consensus termination time of $\Omega(n1.3^n)$; that is, we show that the termination time of consensus may be $\Theta(n)$ times larger than TD even when TD is exponential. To this end, in the graph G_i shown in Figure 1a, we replace the direct edges from R_i to B by a path consisting of processes taken from a set $P \subseteq \{p_{n/2+1}, \ldots, p_n\}$ with $|P| = \Omega(n)$ (i.e., taken away from the original B), as illustrated in Figure 1b.

In more detail, we change the graph construction from Section 6.2 as follows:

- $B = \{n/2+1, \ldots, 0.9n\}$ and $P = \{0.9n+1, \ldots, n\}$;
- 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, according to Corollary 5.

It is easy to see that Claim 15 still holds, as we have $\log_2(N) < 0.4n$, and Claim 16 holds by construction. Regarding Claim 17, 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 19. $G_i^r \sim_p G_{i+1}^r$ for all $r \leq 0.1n$ and all $p \in R_{i+2}$.

Proof. Observe that, by construction, we have $\mathrm{In}_{G_i}(p) = \mathrm{In}_{G_{i+1}}(p)$ 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. ◁

The following Lemma 20 shows that we can even "inflate" arbitrary communication patterns of the oblivious message adversary from Section 6.2:

▶ **Lemma 20.** *Consider $(\sigma, \sigma') \in I(\mathbf{D}^k)$, where $\mathbf{D}$ is the oblivious message adversary of Section 6.2. Let $\tilde{\mathbf{D}}$ be the modified oblivious message adversary of Section 6.3, and $\tilde{\sigma}$ resp. $\tilde{\sigma}'$ in $\tilde{\mathbf{D}}^{(k0.1n)}$ be the communication pattern obtained from replacing every round i graph $\sigma(i)$ resp. $\sigma'(i)$ according to Figure 1a by $0.1n$ instances of the corresponding graph according to Figure 1b. 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 $B = \{p_{n/2+1}, \ldots, p_n\}$. 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 19, we get $\tilde{\sigma}|_{0.1n} \sim_R \tilde{\sigma}'|_{0.1n}$ as needed for (i). As for (ii), the lenght $0.1n$ of the path P in Figure 1b ensures that all processes in B have the same distinguishing power in both the original and in the inflated prefix.

For the induction step $k-1 \to k$, $k > 1$, 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 19 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 Figure 1b, 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 Figure 1b 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 just carries over from $k-1$ to k. ◀

Lemma 20 immediately gives the consensus termination time for our new oblivious message adversary:

▶ **Theorem 21.** *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 18 on the path between G_1^{N-1} and G_2^{N-1} in $I(\mathbf{D})^{N-1}$. As $\mathrm{TD} = N = 1.3^n$, Lemma 13 guarantees that this path exists. Lemma 20 immediately provides us with inflated communication patterns $\mu, \mu' \in I(\mathbf{D})^{0.1n(N-1)}$ for our new oblivious 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 $R_1 = \mathrm{Root}(G_1)$ and $R_2 = \mathrm{Root}(G_2)$ are disjoint, not all processes can have decided by round $0.1n(N-1)$, as claimed. ◀

7 Another Source of Consensus Time Complexity

In this section, we investigate whether the number of iterations TD of the decision procedure is the sole cause for a large time complexity of consensus under an oblivious message adversary. Before we do so, however, let us briefly reiterate what we have achieved so far. In Theorem 12, we have established that consensus can be solved after $c(n-1)\,\mathrm{TD}$ rounds, whereas Theorem 21 revealed that there are in fact oblivious message adversaries where consensus takes up to $n\,\mathrm{TD}$ rounds to terminate and TD may be exponential in n. Thus, in this case, 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 $c(n-1)\,\mathrm{TD}$, and since we have examples where it is at least $n\,\mathrm{TD}$, 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 still exponential in n, i.e., where c is exponentially large. We now proceed to show how to derive such an instance.

7.1 A partition of an oblivious message adversary

Before going into the details of how to construct an oblivious message adversary with the desired property of incurring a large time complexity of consensus while maintaining a low TD, we define 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 $\mathbf{S}_1, \ldots, \mathbf{S}_t$ of $\mathbf{D}$ such that $\mathbf{S}_1$ 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 $\mathbf{S}_2$. Similarly, $\mathbf{S}_2$ is connected in $I(\mathbf{D})$ and all of the edges in this connection, along with the ones from $\mathbf{S}_1$, are protected by the communication graphs of $\mathbf{S}_3$ and so on. Our claim is that if there exist t-round communication pattern that have no common broadcaster and whose round $1 \leq r \leq t$ communication graphs are picked from $\mathbf{S}_r$, then consensus is impossible by round t. The reason for this, as shown in more detail below, is that the set of communication patterns $\mathbf{S}_1 \circ \ldots \circ \mathbf{S}_t$ is connected in the indistinguishability graph $I(\mathbf{D}^t)$, because each $\mathbf{S}_r$ can maintain the connectivity of $\mathbf{S}_1 \circ \ldots \circ \mathbf{S}_{r-1}$ in $I(\mathbf{D}^{r-1})$ as all the edges relevant for this connectivity are protected by the communication graphs of $\mathbf{S}_r$.

Formally, we express this property as follows:

▶ **Definition 22.** *Let $\mathbf{S}_1, \ldots, \mathbf{S}_t$ be a partition of $\mathbf{D}$ with the following properties, for $1 \leq i \leq t$:*

(i) *Each $\mathbf{S}_i$ is connected. That is, for each G, G' in $\mathbf{S}_i$, there is a path from G to G' in the indistinguishability graph $I(\mathbf{D})$ that consists only of elements from $\mathbf{S}_i$.*

(ii) *The edges of the subgraph of $I(\mathbf{D})$, induced by $\bigcup_{j=1}^{i-1} \mathbf{S}_j$, are protected by the communication graphs of $\mathbf{S}_i$.*

(iii) *There is no process p such that every communication pattern of $\Sigma = \mathbf{S}_1 \circ \ldots \circ \mathbf{S}_t$ is broadcastable by p.*

Given this partition, we show in Claim 23 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 23. The communication patterns of $\Sigma_t = \mathbf{S}_1 \circ \ldots \circ \mathbf{S}_t$ are pairwise connected to each other in $I(\mathbf{D}^t)$.

Proof. Throughout this proof, let $I(\mathbf{D})[\mathbf{S}]$ denote the subgraph of $I(\mathbf{D})$, induced by the set of communication graphs $\mathbf{S}$.

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 $e' \in I(\mathbf{D})[\bigcup_{j \leq t} \mathbf{S}_j]$ with the same label $\ell(e) = \ell(e')$. We show this by induction on k with $\Sigma_k = \mathbf{S}_1 \circ \ldots \circ \mathbf{S}_k$.

The base of the induction $k = 1$ follows directly from property (i) of Definition 22, as $\mathbf{S}_1$ 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 $e' \in I(\mathbf{D})[\bigcup_{j \leq k} \mathbf{S}_j]$ 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 $G \in \mathbf{S}_{k+1}$ and G protects e. Such a communication graph G exists because of property (ii) of Definition 22 and because there is an edge $e' \in I(\mathbf{D})[\bigcup_{j \leq k} \mathbf{S}_j]$ 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 22, there is a path π from G to G' in $I(\mathbf{D})$ that consists only of graphs $G'' \in \mathbf{S}_{k+1}$. 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 $I(\mathbf{D})[\mathbf{S}_{k+1}]$ by property (i) of Definition 22. ◁

7.2 An example: choosing the processes

We now construct a set of communication graphs that can be partitioned in accordance with Definition 22, into $t = 1.07^n$ sets. For a set Π of n processes, let $m = \lceil \frac{n}{10} \rceil$. We construct an oblivious message adversary with a partition on it, $\mathbf{D} = \bigcup_{i=1}^{t} \mathbf{S}_i$, where each $\mathbf{S}_i$ is a set of $2i+1$ graphs, denoted $\mathbf{S}_i = \{G_{i,j} \mid 1 \leq j \leq 2i+1\}$. 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 members of the root components of all the graphs $G_{i,j}$; U_i, U_i', with $|U_i| = |U_i'| = m$; and finally $L_{i,j}$, which is the set of all the remaining processes. We choose processes for these sets inductively on i as follows: For the base $i = 1$, we just list the appropriate sets for the communication graphs of $\mathbf{S}_1 = \{G_{1,1}, G_{1,2}, G_{1,3}\}$:

(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) $U_1 \subseteq [2m+1, 4m] \setminus R_3$
(b5) $U_1' \subseteq [1, 2m] \setminus R_2$

We proceed with the inductive step of our construction. For this we assume that we are given $R_1, \ldots, R_{2i+1}$ 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 (but not necessarily disjoint) from $R_2, R_4 \ldots R_{2i+2}$
(s4) We let U_{i+1}' be an arbitrary subset of $[1, 2m]$ of size m, different (but not necessarily disjoint) from $R_3, R_5 \ldots R_{2i+3}$

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.

7.3 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 22 for $t = 1.07^n$ (for an illustration, see Figure 1c). While the choice processes of R_j is independent of i, the edges between them in $G_{i,j}$ will be different and depend crucially on i.

More specifically, the graph $G_{i,j}$ always contains a directed cycle in R_j in increasing order of the process identifiers. Since $|R_j| = m$ and each process already has one incoming edge from the preceding process, there are $m-2$ other potential incoming edges we can choose to add for every member. 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 $\log_2 t < 0.1n < |B|$.

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 $U'_i \cup L_{i,j}$.

7.4 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 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 prove that $\mathbf{D}$ partitions as described in Definition 22.

▷ Claim 24. The sets $\mathbf{S}_1, \ldots, \mathbf{S}_t$ are a partition according to Definition 22.

Proof. First, since all $G_{i,j}$ are different, $S_1, \ldots, S_t$ is indeed a partition of $\mathbf{D}$. For property (i), the connectivity of $\mathbf{S}_i$, pick any $G_{i,j} \in \mathbf{S}_i$. We show that 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 $\mathbf{S}_i$ protect the edges that were used to connect $\mathbf{S}_1, \ldots, \mathbf{S}_{i-1}$, it suffices to show that for every $1 \leq i' < i$, there are communication graphs $G, G' \in \mathbf{S}_i$ such that $\operatorname{Root}(G) \subseteq U_{i'}$ and $\operatorname{Root}(G') \subseteq U'_{i'}$. For a given $1 \leq i' < i$, note that the graphs $G_{i,2i'+2}, G_{i,2i'+3} \in \mathbf{S}_i$ 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 $\Sigma = \mathbf{S}_1 \circ \cdots \circ \mathbf{S}_t$ 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 $1 \leq i \leq t$, $\operatorname{Root}(G_{i,2}) = R_2 \subseteq [1, 2m]$ and $\operatorname{Root}(G_{i,3}) = R_3 \subseteq [2m+1, 4m]$ 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. ◁

▷ Claim 25. The decision procedure terminates after 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 only by processes of B will be preserved after the first iteration. Similarly, when considering the preservation of edges, processes of B in the labels can be ignored.

Now, 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 $\mathbf{D} = \bigcup_{i=1}^t \mathbf{S}_i$ are preserved. Consider $G_{i,j} \in \mathbf{S}_i$, $G_{i',j'} \in \mathbf{S}_{i'}$, $i \neq i'$, 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 $L_{i',j'}$ could intersect ℓ.

Hence, the only processes in ℓ that can occur in a root component of a graph in $\mathbf{D}$ 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 in ℓ and hence 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 $\mathbf{S}_i$ is preserved. Assume for contradiction that for some i, there are graphs $G_{i,j}, G_{i,j'}, G_{i,j''} \in \mathbf{S}_i$, $j \neq j'$ 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 $R_1, \ldots, R_{2i+1}$, which implies $R_{j''} \neq U_i, U'_i$.

If $j = 1$, then the only processes that can appear in ℓ are those of $U_i \cup U'_i$. 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 $\mathbf{S}_i$, and processes of $L_{i,1}$ have incoming edges from R_1, which no process has in any other graphs of $\mathbf{S}_i$. 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 $R_1, \ldots, R_{2i+1}$. 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. ◁

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

▶ **Theorem 26.** *There exists an oblivious message adversary with a consensus time complexity exponential in n in spite of a constant iteration complexity* TD *of the decision procedure.*

8 Conclusions

We presented a simple procedure for deciding whether consensus is solvable 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 worst-case termination time of distributed consensus under a given oblivious message adversary. We discovered a relation between the number of iterations of the decision procedure 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 pursued a combinatorial approach, it would be interesting to study the time complexity of the consensus and other agreement prototols 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. Another interesting avenue of research is a possible algorithmic and/or engineering improvement of our decision procedure, and to empirically evaluate its performance for different oblivious message adversaries.

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. doi:10.1007/978-3-642-35668-1_16.

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. doi:10.1137/16M1081439.

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. doi:10.1007/978-3-642-31104-8_7.

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. doi:10.1016/j.tcs.2018.02.019.

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. URL: http://www.sciencedirect.com/science/article/pii/S0304397510005359.

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. doi:10.1007/978-3-030-24922-9_1.

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 *Lecture Notes in Computer Science*, pages 528–539. Springer Berlin Heidelberg, 2015. doi:10.1007/978-3-662-47666-6_42.

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. doi:10.1007/s00446-009-0084-6.

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. doi:10.1016/j.tcs.2015.01.024.

13 Antoine El-Hayek, Monika Henzinger, and Stefan Schmid. Brief announcement: Broadcasting time in dynamic rooted trees is linear. In *Proceedings of the 2022 ACM Symposium on Principles of Distributed Computing, PODC*, 2022.

14 Antoine El-Hayek, Monika Henzinger, and Stefan Schmid. Asymptotically tight bounds on the time complexity of broadcast and its variants in dynamic networks. In *14th Innovations in Theoretical Computer Science (ITCS)*, 2023.

15 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 2011, Anchorage, Alaska, USA, 16-20 May, 2011 - Conference Proceedings*, pages 1001–1011, 2011. doi:10.1109/IPDPS.2011.96.

16 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.

17 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. doi:10.1145/3212734.3212762.

18 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. doi:10.1016/j.dam.2020.02.013.

19 Eli Gafni. Round-by-round fault detectors (extended abstract): unifying synchrony and asynchrony. In *Proceedings of the Seventeenth Annual ACM Symposium on Principles of Distributed Computing*, pages 143–152, Puerto Vallarta, Mexico, 1998. ACM Press. doi:10.1145/277697.277724.

20 Maurice Herlihy, Dmitry N. Kozlov, and Sergio Rajsbaum. *Distributed Computing Through Combinatorial Topology*. Morgan Kaufmann, 2013. URL: https://store.elsevier.com/product.jsp?isbn=9780124045781.

21 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.

22 Dmitry N. Kozlov. Structure theory of flip graphs with applications to weak symmetry breaking. *CoRR*, abs/1511.00457, 2015. arXiv:1511.00457.

23 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. doi:10.1007/978-3-319-31580-5_7.

24 F. Kuhn and R. Oshman. Dynamic networks: Models and algorithms. *SIGACT News*, 42(1):82–96, 2011.

25 Fabian Kuhn, Nancy A. Lynch, and Rotem Oshman. Distributed computation in dynamic networks. In *STOC*, pages 513–522, 2010. doi:10.1145/1806689.1806760.

26 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.

27 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. doi:10.1177/0037549707085632.

28 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 2019, Toronto, ON, Canada, July 29 - August 2, 2019.*, pages 218–227, 2019. (full version: http://arxiv.org/abs/1905.09590). doi:10.1145/3293611.3331624.

29 Diego Ongaro and John Ousterhout. In search of an understandable consensus algorithm. In *Proc. USENIX Annual Technical Conference (ATC)*, pages 305–319, 2014.

30 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.

31 Nicola Santoro and Peter Widmayer. Agreement in synchronous networks with ubiquitous faults. *Theoretical Computer Science*, 384(2–3):232–249, October 2007.

32 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. doi:10.1137/S009753970443999X.

33 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. doi:10.1145/2833312.2833323.

34 Kyrill Winkler and Ulrich Schmid. An overview of recent results for consensus in directed dynamic networks. *Bulletin of the EATCS*, 128, 2019. URL: `http://bulletin.eatcs.org/index.php/beatcs/article/view/581`.

35 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. `doi:10.4230/LIPIcs.OPODIS.2019.17`.

36 Kyrill Winkler, Manfred Schwarz, and Ulrich Schmid. Consensus in directed dynamic networks with short-lived stability. *Distributed Computing*, 32(5):443–458, 2019. `doi:10.1007/s00446-019-00348-0`.

37 Martin Zeiner, Manfred Schwarz, and Ulrich Schmid. On linear-time data dissemination in dynamic rooted trees. *Discrete Applied Mathematics*, 255:307–319, 2019. `doi:10.1016/j.dam.2018.08.015`.