Efficient Constructions for Almost-everywhere Secure Computation

Siddhartha Jayanti[⋆], Srinivasan Raghuraman[⋆⋆], and Nikhil Vyas[⋆⋆⋆]

MIT

Abstract. The importance of efficient MPC in today's world needs no retelling. An obvious barebones requirement to execute protocols for MPC is the ability of parties to communicate with each other. Traditionally, we solve this problem by assuming that every pair of parties in the network share a dedicated secure link that enables reliable message transmission. This assumption is clearly impractical as the number of nodes in the network grows, as it has today. In their seminal work, Dwork, Peleg, Pippenger and Upfal introduced the notion of *almost-everywhere* secure primitives in an effort to model the reality of large scale global networks and study the impact of limited connectivity on the properties of fundamental fault-tolerant distributed tasks. In this model, the underlying communication network is sparse and hence some nodes may not even be in a position to participate in the protocol (all their neighbors may be corrupt, for instance). A protocol for *almost-everywhere reliable message transmission*, which would guarantee that a large subset of the network can transmit messages to each other reliably, implies a protocol for *almost-everywhere agreement* where nodes are required to agree on a value despite malicious or byzantine behavior of some subset of nodes, and an almost-everywhere agreement protocol implies a protocol *almost-everywhere secure MPC* that is unconditionally or information-theoretically secure. The parameters of interest are the degree d of the network, the number t of corrupted nodes that can be tolerated and the number x of nodes that the protocol may give up. Prior work achieves $d = O(1)$ for $t = O(n/\log n)$ and $d = O(\log^q n)$ for $t = O(n)$ for some fixed constant $q > 1$.

In this work, we first derive message protocols which are efficient with respect to the total number of computations done across the network. We use this result to show an abundance of networks with $d = O(1)$ that are resilient to $t = O(n)$ random corruptions. This randomized result helps us build networks which are resistant to worst-case adversaries. In particular, we improve the state of the art in the almost everywhere reliable message transmission problem in the worst-case adversary model by showing the existence of an abundance of networks that satisfy $d =$

[⋆] jayanti@mit.edu, Supported by the Department of Defense (DoD) through the NDSEG Program, NSF awards CCF-1617730 and IIS-1741137, a Simons Investigator Award, a Google Faculty Research Award, and an MIT-IBM Watson AI Lab research grant.

[⋆⋆] srirag@mit.edu

[⋆⋆⋆] nikhilv@mit.edu, Supported by NSF Grant CCF-1552651.

$O(\log n)$ for $t = O(n)$, thus making progress on this question after nearly a decade. Finally, we define a new adversarial model of corruptions that is suitable for networks shared amongst a large group of corporations that: (1) do not trust each other, and (2) may collude, and construct optimal networks achieving $d = O(1)$ for $t = O(n)$ in this model.

1 Introduction

Many real world applications involve computing functions on large data sets that are distributed across machines in a global network. In many such applications, the data held by any particular agent may need to be kept private. For instance, hospitals across the world have confidential patient data that can be used to create accurate disease models and improve treatment plans. The ubiquitous need for such distributed private computations has motivated research on efficient multiparty computation (MPC) [28][18][3][12]. MPC protocols enable a set of parties to compute a joint function on their inputs while keeping them private [8]. MPC protocols for various important tasks, such as elections, were discovered in the twentieth century, but most of these protocols have not seen practical application as they were designed for densely connected networks and are often inefficient. For MPC to see widespread use, it is important for protocols to rely on only the sparse connectivity that is available in modern large scale networks while simultaneously meeting the efficiency needs of practice. Several avenues of theoretical and practical research remain open: Is it possible to get faster protocols, either through better assumptions, primitives, or special hardware? Is it possible to reduce the amount of communication in the protocol [20][13][7][9]? Is it possible to reduce the number of rounds in the protocol [2]? Is it possible to alleviate the need for synchronicity of nodes in the network? Is it possible to minimize the use of physical resources? Indeed, each of these questions have been considered in the past and remain hotbeds of fruitful research. In this paper, we focus on designing sparse networks and secure communication protocols for these networks that are resilient to large fractions of the machines experiencing byzantine failures, and thereby deviating arbitrarily from the assigned protocols.

All distributed protocols rely on the ability of machines to communicate. In particular, if A and B are two nodes in the network, A must be able to send a message to B in way that satisfies the following two properties: (1) *reliable message transmission*: B receives the message that A intended to send, and (2) *authentication*: B must be able to confirm that A was indeed the sender of the received message [1]. Efficient reliable message transmission is the primary focus of our paper. Reliable transmission becomes trivial if we assume every pair of nodes has a dedicated secure link to pass messages over. However, such an assumption is impractical in modern large scale practical networks which are sparsely connected and rely on multi-hop routing. In a seminal work, Dwork, Peleg, Pippenger and Upfal [16] considered the question of designing sparse networks that are tolerant to nodes experiencing byzantine failures—nodes that fail can deviate arbitrarily from the protocol. The problem is to design a network G of degree d on n nodes in which honest nodes can continue to communicate and execute protocols, even after t nodes are *corrupted*, i.e., experience byzantine failures. The challenge is to make the degree d as small as possible (ideally constant), while letting t become arbitrarily large—ideally εn for some constant ε. Since $t \gg d$, any set of $\Omega(t/d)$ honest nodes can be isolated from the rest of the nodes if all of their neighbors are corrupted. So, it is impossible for all the honest nodes to pairwise communicate with each other. So, apart from the t corrupted nodes, Dwork *et. al.* allow x honest nodes to become *doomed*, and require only the remaining $n - t - x$ *priveleged* nodes to successfully partake in protocols. The class of primitives that work on these privileged nodes in the presence of byzantine failures are called *almost-everywhere (AE)* primitives.

The problem of byzantine agreement [25][23] is one where nodes start with an initial value but wish to agree, at the end of execution of some protocol, on some value, despite malicious or byzantine behavior of some subset of nodes. Prior to [16], this problem was considered assuming all pairs of parties had a dedicated channel for communication [25][23][14]. Dwork *et. al.* introduced the notion of *almost-everywhere agreement* where only privileged nodes need to reach agreement. We note that *AE reliable message transmission*, which would guarantee that a large subset of the network can transmit messages to each other reliably, implies a protocol for *AE agreement*, and an AE agreement protocol implies a protocol for *AE secure MPC* that is unconditionally or information-theoretically secure as formulated in the work of Garay and Ostrovsky [17]. Thus, our goal is to design sparse graphs, and efficient protocols for solving AE reliable message transmission on these graphs.

We consider a network G of degree d on n nodes, and we construct protocols for reliable message transmission in the presence of t corruptions while dooming only x honest nodes. Prior to this work, the focus was to make the degree of the network as small as possible while tolerating a large number t of corrupted nodes and dooming a small number x of honest nodes. We will be concerned about a few more metrics in addition to the degree of the network. We define them below.

1. The *round complexity* of the protocol is the number of rounds of communication required by it.
2. The *node complexity* of the protocol is the number of nodes in the graph that are required in any point-to-point message transmission.
3. The *total work* of the protocol is the number computations performed across all processors in the network in any point-to-point message transmission.

In this work, we obtain results that improve upon all previous works in all parameters. Table 1 shows our main result and compares it to previous works.

Result	**Degree**	**Corruptions**	**Doomed**	**Total Work**	**Node Cmplxty**	**Round Cmplxty**
[16]	$O(1)$	$O(n/\log n)$	$O(t)$	$O(n)$	$O(n)$	$O(n)$
[27]	$O(1)$	$O(n)$	$O(t)$	$O\left(\binom{n}{t}\right)$	$O(n)$	$O(n)$
[10]	polylog(n)	$O(n)$	$O(t/\log n)$	$O(n)$	$O(n)$	$O(n)$
Our work	$O(\log n)$	$O(n)$	$O(t/\log n)$	polylog(n)	polylog(n)	polylog(n)

Our result improves over the most recent work of Chandran *et. al.* in each of the listed metrics. In particular, it is the only work in this line of research to achieve polylogarithmic complexities for the work, node, and round complexity metrics. We believe that these metrics are pivotal for scalability and real-world use. In fact, there results enable us to simulate a protocol on a complete graph with only polylogarithmic multiplicative overhead even though our graph is only of logarithmic degree, while all previous protocols required at least linear multiplicative overhead.

Another approach to obtain more efficient protocols would be to weaken the adversary model. Most prior work has considered worst-case adversarial corrupt nodes. While this is the strongest model one can hope for, it does not necessarily accurately model many real world scenarios. Consider the scenario of a network that is being hacked. Unless an attacker is privy to certain specifics of the network, it is reasonable to assume that the nodes in the network that finally do get hacked are in fact random. For instance, password-guessing attacks or phishing attacks would affect a random subset of the nodes. Ben-or and Ron [4] introduced the *random corruption model* in which nodes are corrupted independently and at random. They exhibited a constant degree network that is resilient to a constant fraction of random corruptions. We show an abundance of graphs with $d = O(1)$ that are resilient to $t = O(n)$ random corruptions while dooming only $x = O(t)$ nodes. Our graphs are significantly simpler than those in the prior work. More significantly, the probability of correctness is substantially large, so large in fact that these constructions help us construct better worst-case networks.

We consider another real world example where the nodes that are trying to perform a secure computational task can naturally be split into different "corporations". A natural model of computation is one where we assume that a certain bounded number of these corporations may be corrupt, while the others are honest. An example of this model is if several companies want to build a secure computation network together, but some of the companies may collude and try to corrupt the network after its topology has been fixed. As another example, the nodes could be routers that are manufactured by different companies, and the adversary could have discovered how to hack into the routers made by some subset of the companies. While in the first example it is reasonable to assume that the network designer knows which nodes belong to which corporation, in the second example even this information is hard to get. Thus, we assume that we neither know which corporations may be malicious nor which nodes belong to which corporations; but we do have a bound on the total number of malicious nodes. We formally define this *corporation corruption model* in Section 2, and give a network topology and protocol that solve this problem near-optimally.

Our contributions

Our first result is a round, node and work efficient protocol.

Theorem. *For the $n = m2^m$-node network $G_{Eff} = (V, E)$ and the protocol Π_{Eff} for message transmission on it there exists constants α and β, such that whp:*

1. *The network G_{Eff} has degree 11.*
2. *The total work is $O(\text{polylog}(n))$.*
3. *The node complexity is $O(\text{polylog}(n))$.*
4. *The round complexity is $O(\log n)$.*
5. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq \alpha n/\log n$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta|T|\log|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{Eff}.*

Our three main results are an optimal protocol in the random model, a protocol worst-case model that improves the current state of the art and a protocol in the corporation model. We present them below.

Theorem (Random Corruptions). *For sufficiently large n, there exists an n-node network $G_{rand} = (V, E)$, a protocol Π_{rand} for message transmission on it, and constants α_3 and β_3, such that:*

1. *The network G_{rand} is of constant degree.*
2. *The total work is $O(\text{polylog}n)$.*
3. *The node complexity is $\text{polylog}(n)$.*
4. *The round complexity is $\text{polylog}(n)$.*
5. *If a subset of nodes $T \subset V$ is randomly corrupt, where $|T| \leq \alpha_3 n$, with probability $1 - 2^{-2t\log(n/t)/\log n}$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta_3|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{rand}.*

Theorem (Worst-case Corruptions). *For the $n = m2^m$-node network $G_{wc} = (V, E)$ and the protocol Π_{wc} for message transmission on it there exists constants α and β, such that whp:*

1. *The network G_{wc} has degree $O(\log n)$.*
2. *The total work is $\text{polylog}(n)$.*
3. *The node complexity is $\text{polylog}(n)$.*
4. *The round complexity is $\text{polylog}(n)$.*
5. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq \alpha n$, there exists a set of nodes $S \subset V$ where $|S| \geq n - |T| - \beta|T|/\log n$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{wc}.*

Theorem (Corporation Corruptions). *For sufficiently large n, there are constants ε, β such that, (G_{corp}, Π_{corp}) is resilient to a constant fraction of corporation-$\log n$ corruptions:*

1. *The network G_{corp} is of constant degree.*
2. *The total work is $O(\text{polylog}n)$.*
3. *The node complexity is $\text{polylog}(n)$.*
4. *The round complexity is $\text{polylog}(n)$.*

5. *If any subset $H \subset [h]$ of corporations is corrupted, i.e., the nodes $T = \cup_{i \in H} V_i$ (for $t = |T| \leq \varepsilon n$) is corrupted, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{corp} with probability $1 - 2^{-t \log(n/t)/\log n}$.*

Our techniques

Round, node and work efficient networks All prior works have $O(n)$ round, node and work complexities (ignoring logarithmic factors). It is easy to see that the diameter of a graph is a lower bound on the round complexity of a protocol. We have no reason to believe this cannot be achieved. We begin by examining the protocol of Dwork *et. al.* [16] over the butterfly network. In this protocol, every pair of nodes has an assigned set of paths of size $\Theta(n/\log n)$ that one floods with the message to be transmitted for the other to recieve and decode using a simple majority. It is the case that for any pair, their paths pass through $O(n)$ many nodes and this immediately makes the round, node and work complexities $O(n)$ if we assume that each node can send a single bit per round. To improve upon this, we show that it is possible to select a much smaller subset of size $O(\log n)$ of those paths for each pair of nodes in the graph that only pass through polylog(n) many nodes. Repeating the protocol from [16], except now restricted to this special small subset of the paths enables us to achieve round, node and work complexities of polylog(n). The part that remains to be seen is what this special subset is. And in fact, we show that many of them work. If one were to sample uniformly at random $O(\log n)$ paths of the $\Theta(n/\log n)$ paths for each pair of nodes and fix these paths ahead of time as the ones to be used in the message transmission protocol, with probability $1 - n^{-t}$, there exists a set of $n - O(t \log t)$ nodes that can reliably communicate with each other in the presence of any adversary that corrupts $t = O(n/\log n)$ nodes. This shows that there exist an abundance of such networks that offer the same degree and resiliency properties as in [16] while also being round, node and work efficient.

Networks resilient to random corruptions The protocol of [10] builds on the following observation. Consider the protocols of [16] and [27] where if node A wishes to communicate with node B, A floods all paths from A to B (possibly of a bounded length) with the message. In [16], the parameters are set to ensure that a majority of such paths contain no corrupt nodes (for most pairs of nodes A, B) while [27] employs an exhaustive search to determine which paths may have contained corrupt nodes. These protocols face the disadvantage that paths that pass through even one corrupt node are lost. The work of [10] introduced the idea of local correction through the use of Bracha committees. If we were able to create committees that had the ability to locally correct the message transmission, we can potentially tolerate a lot more corruptions than in [16] and perform the final decoding more efficiently than in [27]. [10] however considers many overlapping committees in order to ensure that even if a constant fraction of the nodes are corrupt, a sub-constant fraction of the committees are corrupt, where a committee is considered corrupt if a certain fraction of its nodes is corrupt. This calls for a larger degree. In the model of random corruptions, it suffices to construct fewer committees to achieve the same goal. In fact, it is possible to consider non-overlapping committees and since the corruptions are random, if we corrupt a constant fraction ε of the nodes, by a Chernoff bound, most committees are going to be about ε-fraction corrupt. For appropriate ε, this means that most committees in fact work just fine and can perform local correction. Using this observation, we construct a network of constant degree that is round, node and work efficient that tolerates a constant fraction of random corruptions while dooming $O(t)$ nodes.

Logarithmic degree networks in the worst-case model Miraculously, the network resilient to random corruptions paves the way to a rather simple construction of a network of logarithmic degree that is resilient to a constant fraction of worst-case corrupt nodes. The key observation is that the network that is secure is against random corruptions works with an extremely high probability, $1 - 2^{-\tilde{O}(n)}$. This allows to create a new network that is simply a combination of several copies of the network resilient to random corruptions. We then show that several of these copies in fact work perfectly in the presence of any constant fraction of worst-case corrupt nodes and this suffices to obtain a protocol. Once again, this construction works with probability $1 - 2^{-O(n)}$ which shows that there exist an abundance of such networks that have logarithmic degree and are resilient to a constant fraction of worst case corruptions while dooming $O(t)$ nodes. This protocol also turns out to be round, node and work efficient. This gives us our main result–a protocol that improves the state of the art in terms of all parameters.

Networks secure in the corporation model We consider the setting in which the nodes that are part of the network belong to different corporations, each of which owns a non-negligible share of the nodes of the network. In the standard adversarial corruption model, each individual node can become corrupt independently, and thus t corruptions can be realized as an arbitrary subset of size at most t nodes becoming corrupt. In the corporation corruption model, we assume that corruptions happen at the level of the corporation, and thus that if a corporation becomes corrupt, all of its nodes are corrupted. So, t corruptions can be realized in this model as an adversarial collection of corporations whose sizes sum up to at most t. If corporations could be arbitrarily small, and simply own single nodes in the network, this model is precisely equivalent to the worst-case adversarial corruption model. The model becomes interesting, when the size of the smallest corporation is $\Omega(f(n))$, for some (slow growing) function f. We therefore define the parametrized corruption model to be the described corruption model, when the smallest corporation owns at least $f(n)$ nodes. We show the abundant existence of constant degree network that is resilient to a constant fraction of nodes being corrupted in the corporation-$\log(n)$ model. We look back to our constant degree network that is resilient to a constant fraction of random corruptions. An arbitrary partitioning of the nodes into non-overlapping committees sufficed to be resilient to random corruptions. Corporation corruptions are non-random, and in fact adversarial. We show however, the surprising result that essentially the same network will achieve resilience to corporation-$\log n$ corruptions, even when a constant fraction of the nodes are being corrupted. The result is achieved by inserting more randomness into the structure, leveraging the extremely high probability of success in the random model construction, and applying, yet again, the probabilistic method. Unsurprisingly, this protocol is also of constant degree and is round, node and work efficient.

Related work

There have been a plethora of works asking for various different measures of quality of an agreement or MPC protocol. A sequence of works seek to improve the round complexity of protocols for byzantine consensus [5][6]. Another goal is to optimize the communication complexity of byzantine agreement protocols [15][22][21][19]. Another model of corruptions is that of edge corruptions [11]. As observed in the work of Chandran *et. al.*, an almost-everywhere secure computation protocol for node corruptions can be readily transformed into a corresponding almost-everywhere protocol also tolerating edge corruptions, for a reduced fraction of edge corruptions (by a factor of d, the degree of the network). We note that all our results hence also extend to the edge corruption model, both worst-case and random.

Organization

We discuss preliminary notations and definitions in Section 2. In Section 3, we present the construction of a network of constant degree that has logarithmic round complexity polylogarithmic work and node complexities. In Section 5, we present the construction of a network achieving $d = O(\log n)$ while handling $t = O(n)$. In Sections 4 and 6, we present our results in the random and corporation models respectively.

2 Preliminaries

2.1 Notation

For $n \in \mathbb{N}$, let $[n] = \{1, 2, \dots, n\}$. We assume that all logarithms are taken to the base 2.

2.2 Chernoff Bounds

Let X be a random variable with $\mathbb{E}[X] = \mu$. For all $\delta \geq 0$,

$$\Pr[X \geq (1+\delta)\mu] \leq \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\mu} \tag{1}$$

$$\Pr[X \le (1-\delta)\mu] \le \left[\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{\mu} \tag{2}$$

Observing that $\ln(1+\delta) \ge \frac{2\delta}{2+\delta}$ for all $\delta \ge 0$, we have for all $\delta \ge 0$,

$$\Pr[X \ge (1+\delta)\mu] \le e^{-\frac{\delta^2\mu}{2+\delta}} \tag{3}$$

$$\Pr[X \le (1-\delta)\mu] \le e^{-\frac{\delta^2\mu}{2+\delta}} \tag{4}$$

Furthermore, for $0 \le \delta \le 1$,

$$\Pr[X \ge (1+\delta)\mu] \le e^{-\frac{\delta^2\mu}{3}} \tag{5}$$

$$\Pr[X \le (1-\delta)\mu] \le e^{-\frac{\delta^2\mu}{2}} \tag{6}$$

2.3 Expanders and Compressors

Definition 1. *A graph $G = (V, E)$ is an expander if there exists a constant $\theta < 1$ such that for every subset $U \subset V$ of vertices of size $|U| \le \frac{|V|}{2}$, the set of vertices outside U that have at least one neighbor in U is at least $\theta|U|$.*

Constructions of expanders of constant degree are known [24].

Definition 2. *A graph $G = (V, E)$ is a compressor if there exists a constant $\theta < 1$ such that for every subset $U \subset V$ of vertices of size $|U| \le \theta|V|$, the set of vertices that have at least half of their neighbors in U is at most $\frac{|U|}{2}$.*

Constructions of compressors of constant degree are known [26][24][16].

2.4 Network Parameters

Given a graph $G = (V, E)$, a *message transmission protocol* or simply *protocol* Π on the graph, is a specification for how messages are routed between every pair of nodes. In particular, $\Pi(u, v)$ is the protocol for node $u \in V$ to *transmit* to node $v \in V$. A protocol is comprised of *rounds*. In each round, we allow each node $w \in V$ to perform local computations and pass a different one bit message to each of its neighbors in G.

We call a pair $N = (G, \Pi)$ a *protocoled-network* if Π is a protocol for graph G. We define the following properties of the network, where u and v are two different nodes in G:

1. **Round complexity:** The round complexity of $\Pi(u, v)$ is the number of rounds, $r(u, v)$, required by $\Pi(u, v)$. The *round complexity* of Π is $r \triangleq \max_{u,v\in V} r(u, v)$.
2. **Node complexity:** The node complexity of $\Pi(u, v)$ is the number of nodes, $\nu(u, v)$, in the graph that are required in a transmission from u to v. The *node complexity* of Π is $\nu \triangleq \max_{u,v\in V} \nu(u, v)$.

3. **Work complexity**, or, **Total work:** The total work of $\Pi(u, v)$ is the number computations, $W(u, v)$, performed across all processors in the network in a transmission from u to v. The *total work* of Π is $W = \max_{u,v \in V} W(u, v)$.
4. **Graph degree:** The degree of u is the number of neighbors, $d(u)$, that u has in G. The *degree* of G is $d = \max_{u \in V} d(u)$.
5. **Resilience:** We say the network is *resilient* to a set of nodes T, of size $t = |T|$, being *corrupted* while dooming only x nodes if there is a subset $S \subseteq V$ of $n - t - x$ *privileged* nodes that can reliably transmit messages between each other, after the nodes in T experience byzantine failure. Nodes in set S are called *privileged*, nodes in $X = V - (S \cup T)$ are called *sacrificed*, and nodes in $X \cup T$ are called *doomed*. Informally speaking, a network is highly resilient if even when t is large, x is not too large, and thus $|S|$ is large.

Our goal is to design highly resilient low degree networks of low round, node, and work complexity.

Remark 1. All our networks have node and work complexities that differ by at most a polylogarithmic factor. This is also true of almost all the previous works we mention. A notable exception is Upfal's [27] network on an expander, which requires exponential work.

2.5 Notion of Almost-everywhere Security

The notion of almost-everywhere secure primitives was introduced by Dwork *et. al.* [16]. In this setting, we consider a sparse communication network on the nodes. We assume a synchronous network and that the communication is divided into rounds. In every round, each node can send (possibly different) messages on its incident edges; these messages are delivered before the next round. Suppose a certain subset of the nodes may be adversarially corrupt, in particular adaptive, rushing and computationally unbounded. This implies that a protocol for any task on this network must “give up” a certain number of honest nodes on account of their poor connectivity to other honest nodes. We set up the following notation. Consider a network of n nodes connected by a communication network $G = (V, E)$ of degree d. On executing a protocol Π on this network in the presence of a subset $T \subset V$ of adversarial or *corrupt* nodes, let $X \subset V$ be the set of honest nodes that are given up, or *doomed*, and let $P \subset V$ be the set of honest nodes for whom the protocol requirements of correctness and security hold, or *privileged* nodes. The nodes that are not privileged are *unprivileged* nodes. Let $|T| = t$, $|X| = x$ and $|S| = s$. We have $t + x + s = n$.

2.6 Almost-everywhere Reliable Message Transmission

Definition 3. *A protocol Π for almost-everywhere reliable message transmission on an n-node network $G = (V, E)$ is one that satisfies the following requirement: If a subset of nodes $T \subset V$ is corrupt, there exists a set of nodes $S \subset V$*

such that every pair of nodes in S can communicate reliably with each other by invoking Π.

We present prior protocols for almost-everywhere reliable message transmission.

Dwork, Peleg, Pippenger, Upfal [16] Dwork *et. al.* define the *butterfly* protocol-network.

Definition 4. *The butterfly network (G_{But}, Π_{But}) is as follows.*

Graph*: $G_{But} = (V_{But}, E_{But})$ where $V_{But} = \{(i,j)\}$ where $0 \leq i \leq m-1$ and $j \in \{0,1\}^m$ is a set of $n = m2^m$ nodes, and $E_{But} = \{(i,j),(i',j')\}$ is the set of edges where $i' = (i+1) \mod m$ and j and j' only possibly differ in the i^{th} bit.*

Protocol*: Let u and v be distinct vertices in V_{But}. There exists as set of paths $P_{u,v}$ from u to v such that $|P_{u,v}| = 2^m = \Theta(n/\log n)$. The message transmission protocol Π from u to v in G_{But} is as follows: u sends the message along all paths $P_{u,v}$, v receives all the messages and takes majority.*

Theorem 1 ([16]). *For the $n = m2^m$-node network $G_{But} = (V,E)$ and the protocol Π_{But} for message transmission on it, there exists constants α_1 and β_1, such that:*

1. *The network G_{But} is of constant degree, namely 11.*
2. *The node complexity is $\tilde{O}(n)$.*
3. *The round complexity is $\tilde{O}(n)$.*
4. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq \alpha_1 n/\log n$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta_1 |T| \log |T|$ such that for every pair of nodes (u,v) in S, $(2/3)^{rd}$ of the paths in $P_{u,v}$ have no corrupted nodes in them which implies that all pairs of nodes in S can communicate reliably with each other by invoking Π_{But}.*

Within their construction, Dwork *et. al.* describe a compressor network $G = (V,E)$ of constant degree 7 and an associated compression procedure which works by running the following procedure for $\log |V|$ rounds:

1. Each node sends its value to all its neighbours.
2. Each node receives the values of all its neighbors.
3. Each node chooses as its new value the value held by majority of its neighbors.

If the number of corrupt nodes is $t \leq \frac{\theta|V|}{2}$ for some constant θ, we have the following lemma [16].

Lemma 1. *If there are at least $(1-\theta)n$ honest processors which share the same initial value v, then after applying the compression procedure, at most $t+1$ honest nodes will have a value different from v.*

Upfal [27]

Theorem 2. *For sufficiently large n, there exists an n-node network $G_{Upfal} = (V, E)$, a protocol Π_{Upfal} for message transmission on it, and constants α_2 and β_2, such that:*

1. *The network G_{Upfal} is of constant degree*[1]*.*
2. *The number of rounds of communication in Π_{Upfal} is $O(\log n)$.*
3. *The local computation time of a node in Π_{Upfal} is $O(2^n)$.*
4. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq \alpha_2 n$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta_2|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{Upfal}.*

2.7 Almost-everywhere Agreement

Definition 5. *A protocol Π for almost-everywhere agreement on an n-node network $G = (V, E)$, where party P_i holds value v_i for $i \in [n]$, is one that satisfies the following requirement: If a subset of nodes $T \subset V$ is corrupt, there exists a set of nodes $S \subset V$ such that by invoking Π:*

1. *All nodes in S output the same value.*
2. *If $v_i = v$ for all $i \in S$, then all nodes in S output v.*

Protocols for traditional byzantine agreement assume a fully connected network and require that all honest nodes reach agreement, while in almost-everywhere agreement, we only require that the privileged nodes reach agreement.

Theorem 3. *Let Π_{BA} be a protocol for traditional byzantine agreement that tolerates up to $t+x$ corrupt nodes, and let Π_{TS} be a protocol for almost-everywhere reliable message transmission on a network G of degree d that in the presence of t corrupt nodes gives up x nodes. Then, there exists a protocol Π for almost-everywhere agreement on G such that:*

1. *The number of rounds of communication in Π is the product of the number of rounds of communication in Π_{BA} and Π_{TS}.*
2. *The local computation time of a node in Π is the product of the number of rounds of communication in Π_{BA} and Π_{TS}.*
3. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq t$, there exists a set of nodes $S \subset V$ where $|S| \geq n - t - x$ that achieve agreement as in Definition 5.*

[1] G_{Upfal} is an n node Ramanjuan graph, and we know such graphs with large enough constant degree.

2.8 Almost-everywhere Secure Computation

Garay and Ostrovsky [17] showed that if one has a protocol for almost-everywhere reliable message transmission on a network G, one can obtain a protocol for secure (private) and reliable message transmission over a network G' whose degree is more than that of G by only a constant factor. Furthermore, G' preserves the set S of privileged nodes in G asymptotically. This allows us to obtain protocols for almost-everywhere secure computation.

Theorem 4. *Let Π_{MPC} be a protocol for multiparty computation that tolerates up to $t + x$ corrupt nodes, and let Π_{TS} be a protocol for almost-everywhere reliable message transmission on a network G of degree d that in the presence of t corrupt nodes gives up x nodes. Then, there exists a graph G' and a protocol Π for almost-everywhere secure computation on G' such that:*

1. *The degree of the network G' is a constant times the degree of the network G.*
2. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq t$, there exists a set of nodes $S \subset V$ where $|S| \geq n - t - x$ that achieve agreement as in Definition 5.*

2.9 Corruption Models

We consider many models where a subset T of size t in the n node network can be corrupted.

Worst-case Model The worst-case model is the strongest of our adversary models. In this model, the subset of T corrupt nodes can be chosen adversarially after the network topology and protocol for communication have been fixed.

Random Model The randomized adversary model assumes that the t corrupted nodes are chosen uniformly at random from the set of n nodes. We call this model of picking a random subset of size t the Hamming Random Model or corruption. Alternately, a randomized adversary may make each node corrupt with probability t/n; we call this the Shannon model. Basic Chernoff bounds show that the Shannon and Hamming models are equivalent up to a constant factor difference in t with all but exponentially small probability. Thus, we freely switch between the two models in our exposition. While this model of corruption is primarily good for simulating phishing and password guessing attacks, our probabilistic approaches show that it can be the starting point for state of the art protocols against corporation and worst-case adversaries.

Corporation Model We consider the setting in which the n nodes that are part of the network belong to k different corporations, each of which owns a non-negligible share of the nodes of the network. That is V is the disjoint union of $V_1, \ldots, V_k$, where the nodes of each V_i belong to the ith corporation. We do not assume that we know the partition. In the standard adversarial corruption model, each individual node $i \in V$ can become corrupt independently, and thus an ε-fraction of corruptions can be realized as an arbitrary subset of size at most εn nodes becoming corrupt. In the corporation corruption model, we assume that corruptions happen at the level of the corporation, and thus that if the ith corporation becomes corrupt, all the nodes of V_i are corrupted. So, an ε-fraction of corruptions can be realized in this model as an adversarial collection of corporations $K \subset [k]$ becoming corrupt, where $\left|\bigcup_{i \in K} V_i\right| \leq \varepsilon n$. If corporations could be arbitrarily small, and simply own single nodes in the network, this model is precisely equivalent to the worst-case adversarial corruption model. The model becomes interesting, when the size of the smallest corporation is $\Omega(f(n))$, for some (slow growing) function f. We therefore define the parametrized corruption model *corporation-$f(n)$* to be the described corruption model, when the smallest corporation owns at least $f(n)$ nodes. This parametrized model is thus an adversarial model, that smoothly moves from the trivial corporation-n to the worst-case corruption model, corporation-1. We show the abundant existence of constant degree network that is resilient to a constant fraction of nodes being corrupted in the corporation-$\log(n)$ model.

3 Low-work Protocols in the Worst-case Model

It is our goal to design low degree graphs with efficient communication protocols for AE reliable message transmission. Our final networks are constructed by composing several simpler graph structures. An important graph that our work builds on is Dwork *et. al.*'s *butterfly* network [16]. The diameter of a graph is a fundamental lower bound on the number of rounds required for message transmission. Any graph with constant degree will necessarily have a diameter of length $\Omega(\log n)$. Thus, the logarithmic diameter of the butterfly network is optimal up to constant factors. It is thus reasonable to consider *fast* message transmission as on which requires a polylogarithmic number of rounds. However, Dwork *et. al.*'s protocol requires $\Omega(n)$ rounds for a single point to point message transmission. (This can be improved to polylog(n) only if each process is allowed to transmit $O(n)$ bits per round.)

Their protocol is also very inefficient in another sense. Dwork *et. al.*'s message transmission protocol requires nearly every node of the graph to participate in every message transmission. This is undesirable for two reasons:

1. *High node complexity*: It is non-ideal to make every node of a large network participate in every point to point message transmission. It would aid both efficiency and parallelizability of higher level protocols to limit the number of nodes used for a point to point transmission to $O(\text{polylog}(n))$.
2. *High work complexity*: For a single A to B transmission, the total work is $\Omega(n)$, which is exponentially large in the diameter (which is the fundamental lower bound on total work).

We therefore design two transmission protocols that ameliorate these flaws. To get a protocol that is work and node efficient, we modify the protocol of Dwork *et. al.* in the following way. We run over the same butterfly graph, but show that we need not flood all $\Theta(n/\log n)$ paths in the network to ensure reliable transmission, but rather need to pick a set of $\Theta(\log n)$ paths between every pair of vertices. This reduces both the number of nodes used per point to point transmission and total work to $O(\log^2 n)$.

Definition 6.

> **Graph***: $G_{Eff} = (V, E) = G_{But}$ as defined in Definition 4 such that $|V| = n = m2^m$. For every pair u, v of distinct vertices in V, there exists a set of paths $P_{u,v}$ as defined in Definition 4 between u and v. Let $Q_{u,v}$ be a random subset of $P_{u,v}$ of size $\Theta(m)$. The subset $Q_{u,v}$ is sampled before the protocol and is fixed, in particular it is known to all the nodes as well as the adversary.*
> **Protocol***: The message transmission protocol Π from u to v in G_{Eff} is as follows: u sends the message along all paths in $Q_{u,v}$, v receives all the messages and takes majority. We call this protocol Π_{Eff}.*

Theorem 5. *For the $n = m2^m$-node network $G_{Eff} = (V, E)$ and the protocol Π_{Eff} for message transmission on it there exists constants α and β, such that whp:*

1. *The network G_{Eff} has degree 11.*
2. *The total work is $O(\text{polylog}(n))$.*
3. *The node complexity is $O(\text{polylog}(n))$.*
4. *The round complexity is $O(\log n)$.*
5. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq \alpha n/\log n$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta|T| \log |T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{Eff}.*

Proof. It is clear that the degree of the network is 11 and that the node complexity and the total work in the protocol are $O(\text{polylog}(n))$. To achieve $O(\log n)$ round complexity we will send messages along the i^{th} path in the i^{th} round. By construction of the butterfly graph, this will ensure that there is no congestion due to messages from separate paths.

Consider any fixed subset $T \subset V$ with $t = |T| \leq \alpha n/\log n$. By Theorem 1, for appropriate constants α, β, we know that there is a set V' of size $n - \beta t \log t$ that can communicate reliably with each other by invoking Π_{But}. For any pair of vertices $u, v \in V'$, we let $P_{u,v}$ be the set of paths used in message transmissions from u to v by protocol Π_{But}. From Subsection 2.6 we know that at least a 2/3 fraction of the paths in each $P_{u,v}$ contain no corrupt node. Let $Q_{u,v}$ be a random sample of $100 \log n$ paths from $P_{u,v}$. The protocol Π_{Eff} to send a message from u to v is as follows: (1) u sends the message along all the paths $Q_{u,v}$, (2) v receives all $100 \log n$ messages that were sent along the paths in $Q_{u,v}$ and takes the majority.

We first argue correctness of our protocol. For a fixed adversary and for fixed $u, v \in V'$, the probability that a majority of the paths $Q_{u,v}$ contain a corrupt node is $\leq 1/n^4$ by a Chernoff bound. We call a pair of vertices $\{u, v\}$ *corrupted* if a majority of paths between them contain a corrupt node, and *uncorrupted* otherwise. So by another Chernoff bound, for a fixed adversary, the probability that there are more than t corrupted pairs is bounded above by

$$\binom{n^2}{t}\left(\frac{1}{n^4}\right)^t < \left(\frac{1}{n^2}\right)^t$$

since the probability of pair corruptions is independent conditioned on the adversary. To show that the construction works for an adversarially chosen set of corruptions, we take a union bound over all adversaries. The probability that there is an adversary for which the number of corrupt pairs is at least t is bounded above by:

$$\binom{n}{t}\left(\frac{1}{n^2}\right)^t < \left(\frac{1}{n}\right)^t \ll 1$$

So, our network satisfies the part (5) of the theorem as well. Since the probability is $\ll 1$ any graph (with the paths) that we sample will work with high probability.

□

Theorem 5 shows that the graph G_{Eff} satisfies properties (1-5) with high probability. The following corollary follows immediately by the probabilistic method; yet we state it explicitly, because we will use the graph with these properties in our subsequent constructions.

Corollary 1. *There is a fixed graph G_{Eff}, and protocol Π which satisfy the conditions (1-5) of Theorem 5.*

4 Constant-degree Networks in the Random Model

In the previous section, we constructed networks that have low degree and are also node and work efficient. However, they only tolerate a $1/\log n$-fraction of corrupted nodes and $O(t \log t)$ nodes are doomed. We now wish to improve these fault tolerance parameters. We turn our attention to the protocol of [10]. Their protocol builds on the following observation. Consider the protocols of [16] and [27] where if node A wishes to communicate with node B, A floods all paths from A to B (possibly of a bounded length) with the message. In [16], the parameters are set to ensure that a majority of such paths contain no corrupt nodes (for most pairs of nodes A, B) while [27] employs an exhaustive search to determine which paths may have contained corrupt nodes. These protocols face the disadvantage that paths that pass through even one corrupt node are lost. The work of [10] introduced the idea of local correction through the use of Bracha committees. If we were able to create committees that had the ability to locally correct the message transmission, we can potentially tolerate a lot more corruptions than in [16] and perform the final decoding more efficiently than in [27]. [10] however considers many overlapping committees in order to ensure that even if a constant fraction of the nodes are corrupt, a sub-constant fraction of the committees are corrupt, where a committee is considered corrupt if a certain fraction of its nodes is corrupt. This calls for a larger degree. We show in this section that in our model of random corruptions, it suffices to construct fewer committees to achieve the same goal. Going forward, we refer to the networks (protocol, resp.) of [27] by G_{Upfal} ($\mathit{\Pi}_{Upfal}$ resp.) respectively.

Let the set of nodes that wish to communicate be $V = [n]$ for $n \in \mathbb{N}$. We arbitrarily divide the nodes of V into n/s committees of size $s = \log\log n$. Within each committee, we instantiate G_{Upfal}, which is an expander of constant degree $d = O(1)$. We then connect the n/s committees using the network G_{Eff} from the previous section, where in order to connect two committees, we connect them by means of a perfect matching between the two sets of s nodes.

Definition 7.

Graph: *Our graph that is resistant to random errors is $G_{rand} = (V, E)$, where $V = [n]$. The edge set is as follows. Arbitrarily partition the nodes of V into n/s committees of size $s = O(\log\log n)$. We let C_v denote the committee containing node v, where $C_u = C_v$ if u and v are in the same committee. Within each committee, we instantiate G_{Upfal}, which is an expander of constant degree $d = O(1)$. We then connect the n/s committees using the G_{Eff}, where in order to connect two committees, we connect them by means of a perfect matching between the two sets of s nodes.*

Protocol: *We now describe the communication protocol $\mathit{\Pi}_{rand}$ over this network. To this end, we first describe two building block protocols $\mathit{\Pi}_{edge}$ and $\mathit{\Pi}_{maj}$.*

- *$\mathit{\Pi}_{edge}$ is the protocol that is invoked when we wish to send a message from one committee, C to another C' that are connected in the G_{Eff} network (connected by means of a perfect matching). We will assume that each node in C is initialized with some message. In the protocol $\mathit{\Pi}_{edge}$, each node in C sends its message to the node it is matched to in C'.*
- *$\mathit{\Pi}_{maj}$ is a majority protocol invoked within a committee C. We will assume that each node i in C is initialized with some message m_i. The goal of the $\mathit{\Pi}_{maj}$ protocol is for each node in C to compute the majority function $m = \mathrm{maj}\{m_i\}_i$. The protocol proceeds as follows: every node in C invokes $\mathit{\Pi}_{Upfal}$ to send its message to every other node in C. Each node then simply computes (locally) the majority of the messages it received.*

Now, if a node A wishes to send a message m to node B:

(a) If A and B are in the same committee C, then A simply sends the message to B by invoking $\mathit{\Pi}_{Upfal}$ within the committee C.

(b) If A and B are in different committees, C_A and C_B respectively, then:

   i. A invokes $\mathit{\Pi}_{Upfal}$ to send m to every other node in its committee C_A.

   ii. The committee C_A then invokes $\mathit{\Pi}_{Eff}$ to send a message to the committee C_B. In the invocation of $\mathit{\Pi}_{Eff}$, whenever two committees C

and C' connected by G_{Eff} wish to communicate with each other, they invoke Π_{edge} and then C' invokes Π_{maj}.

iii. Finally, every node other than B in committee C_B invokes Π_{Upfal} to send the message they received to B. B computes (locally) the majority of the messages it received.

Lemma 2. *The network constructed is of constant degree, namely $D = d + 11$.*

Lemma 3. *Protocol Π_{rand} has* $\operatorname{polylog}(n)$ *round complexity and* $\operatorname{polylog}(n)$ *work complexity.*

Proof. The subprotocol Π_{maj} takes $O(2^s) = \operatorname{polylog}(n)$ rounds and work by construction of (G_{Upfal}, Π_{Upfal}) protocol [27]. Π_{edge} clearly takes $O(1) = \operatorname{polylog}(n)$ rounds and work. Since the diameter of G_{Eff} is $O(\log n) = \operatorname{polylog}(n)$, and polylogarithms are closed under addition, multiplication, and composition, Π_{rand} has $\operatorname{polylog}(n)$ round and work complexity. □

We now wish to argue that in the presence of a set $T \subset V$ of randomly corrupt nodes with $|T| \leq \alpha_3 n$, there exists a set $S \subset V$ with $|S| \geq n - \beta_3 |T|$ such that every pair of nodes in S can communicate reliably with each other, for constants α_3, β_3. The proof proceeds as follows. We first show that most committees must in fact contain close to an α_3-fraction of corrupt nodes; call such committees as *good* committees. For appropriately chosen parameters, this would ensure that Π_{Upfal} works successfully for all but an ϵ-fraction of nodes in these *good* committee, for some small ϵ. We now consider nodes A, B that wish to communicate with each other, and are also in *good* committees. Since A is in a *good* committee, all but an ϵ-fraction of the nodes in the committee receive A's message correctly. On any execution of Π_{edge} between *good* committees, all but at most an ϵ-fraction of the nodes in the receiving committee receive the correct value. Now, in an execution of the Π_{maj} protocol, all but at most a 2ϵ-fraction of the nodes begin with the correct value and only and Π_{Upfal} works successfully for all but an ϵ-fraction of nodes. This ensures that as long as $\epsilon < 1/4$, all but at most an ϵ-fraction of the nodes compute the majority of the incoming messages correctly. Inductively, this would show that at the end of the emulation of the Π_{Eff} protocol, all but an ϵ-fraction of the nodes in the committee containing B receive A's message correctly and since B is in a good committee and $\epsilon < 1/4$, B receives A's message correctly.

We now formalize this argument. We call a committee *good* if the fraction of corrupt nodes in it is at most α_2 and *bad* otherwise. Let $T \subset V$ be a set of randomly corrupt nodes with $|T| = t \leq \alpha_3 n$ where $\alpha_3 = \alpha_2/100$ where the constant α_2 is from Theorem 2.

Lemma 4. *The probability that a committee is* good *is at least* $1-(t/n)^{10 \log \log n}$.

Proof. Using the Chernoff bound from (5), we have the probability that a committee is bad is at most $e^{-\left(\frac{\alpha_2}{\alpha_3}-1\right)^2 \frac{\alpha_3 s}{3}} \leq (t/n)^{10 \log \log n}$. □

We have that if C is a *good* committee with $t' \leq \alpha_2 s$ corrupt nodes, from Theorem 2, there exists a set S_C (privileged nodes) of at least $s - \beta_2 t'$ nodes in C that can communicate reliably with each other.

Lemma 5. *The number of* bad *committees is at most* $\frac{t/s}{\log(n)}$ *with probability at least* $1 - 2^{-2t\log(n/t)/\log n}$.

Proof. Using the Chernoff bound from (1), we have the probability that the number of bad committees is more than $\frac{t/s}{\log(n)}$ is at most $e^{-\frac{\delta \log \delta\zeta n}{2s}}$ where $\zeta = (t/n)^{10\log\log n}$ and $\delta = (t/n)/((t/n)^{10\log\log n}\log n)$. On simplification, this gives $e^{-2t\log(n/t)/\log n}$. □

We say that a committee holds value v if all the privileged nodes in the committee hold value v.

Lemma 6. *If C and C' are* good *committees connected by an edge in G_{Eff} and if C holds value v, after invoking Π_{edge} and Π_{maj}, C' holds value v.*

Proof. Since C holds value v, at least $s - \beta_2\alpha_2 s$ nodes in C' receive the value v after invoking Π_{edge}. Hence, at least $s - (\beta_2 + 1)\alpha_2 s$ nodes in C' begin with the value v while invoking Π_{maj} in C'. Consider the set $S_{C'}$ of privileged nodes in C'. We have $|S_{C'}| \geq s - \beta_2\alpha_2 s$. They receive messages reliably from each other. Of these messages, at most $(\beta_2 + 1)\alpha_2 s$ may be unequal to v. Thus each node in $S_{C'}$ will receive at least $s - (2\beta_2 + 1)\alpha_2 s$ copies of v. Hence, if $(2\beta_2 + 1)\alpha_2 < 1/2$, the claim follows. We note from [27] that it is possible to take $\alpha_2 = 1/72$ and $\beta_2 = 6$ which satisfies $(2\beta_2 + 1)\alpha_2 < 1/2$. □

Considering the *bad* committees as corrupt nodes in G_{Eff}, since there are at most $\frac{t/s}{\log n}$ of them (with overwhelming probability), from Theorem 5, there exists a set of committees P (privileged committees) that can communicate with each other reliably.

Lemma 7. *Let A and B be two nodes in privileged* good *committees $C_A \in P$ and $C_B \in P$ respectively. If $A \in S_{C_A}$ and $B \in S_{C_B}$, then the above protocol guarantees reliable message transmission from A to B.*

Proof. Note that if $C_A = C_B$, we are done. We consider the case $C_A \neq C_B$. Since $A \in S_{C_A}$, all nodes in S_{C_A} receive A's message, m, correctly and C_A holds m. Since $C_A, C_B \in P$, after the invocation of Π_{Eff}, C_B holds m. Since $B \in S_{C_B}$, it receives m from each node in $S_{C'}$. Hence B will receive at least $s - \beta_2\alpha_2 s$ copies of v. If $\beta_2\alpha_2 < 1/2$, the claim follows. We note from [27] that it is possible to take $\alpha_2 = 1/72$ and $\beta_2 = 6$ which satisfies $\beta_2\alpha_2 < 1/2$. □

Lemma 8. *With probability $1 - 2^{-2t\log(n/t)/\log n}$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta_3|T|$ such that every pair of nodes in S can communicate reliably with each other.*

Proof. The set S consists of nodes that are privileged nodes in privileged *good* committees. Let t'' denote the number of bad committees. Note that with probability at least $1-2^{-2t\log(n/t)/\log n}$, $t'' \leq \frac{t/s}{\log n}$. From Theorem 5, number of nodes in unprivileged committees is bounded by $O(st''\log t'') = O(t)$. Finally, we consider the unprivileged nodes in privileged committees. Let t_i denote the number of corrupt nodes in committee C_i for $i \in [n/s]$. The number of unprivileged nodes in privileged committees is upper bounded by

$$\sum_i O(t_i) = O\left(\sum_i t_i\right) = O(t)$$

from Theorem 2. Thus, $|S| \geq n - O(t)$. □

We summarize the result from this section in the theorem below.

Theorem 6. *For sufficiently large n, there exists an n-node network $G_{rand} = (V, E)$, a protocol Π_{rand} for message transmission on it, and constants α_3 and β_3, such that:*

1. *The network G_{rand} is of constant degree.*
2. *The total work is $O(\text{polylog}n)$.*
3. *The node complexity is $\text{polylog}(n)$.*
4. *The round complexity is $\text{polylog}(n)$.*
5. *If a subset of nodes $T \subset V$ is randomly corrupt, where $|T| \leq \alpha_3 n$, with probability $1 - 2^{-2t\log(n/t)/\log n}$, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta_3|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{rand}.*

We end this section with the following remark. Let $|T| = t$. Note that in [10], the number of nodes that can communicate with each other reliably is $n - t - O(t/\log n)$, that is, we give up at most $O(t/\log n) = o(t)$ nodes. We remark that this is not achievable in networks of constant degree even in the random model. In an adversarial corruption setting, one can corrupt the neighbors of $O(t/d)$ nodes, and hence if $d = O(1)$, any protocol must give up $O(t)$ nodes. This is true even in the random corruption model: a node has corrupt neighbors with some constant probability if $t = O(n)$ and hence any protocol must give up $O(t)$ nodes. Similarly, in networks of $\log\log n$ degree, any protocol must give up $O(t/\log n)$ nodes.

5 Logarithmic degree Networks in the Worst-case Model

In the worst-case model, the current best networks are those constructed by by Chandran, Garay and Ostrovsky [10]. They construct a graph with degree $d = \log^q n$ for some fixed constant $q > 1$, that is resilient to $t = O(n)$ adversarial corruptions. We show using a probabilistic argument the existence of a network of degree $O(\log n)$ graph that is resilient to $t = O(n)$ adversarial corruptions.

Furthermore, the probabilistic construction works with all but negligibly small probability.

Our construction is also rather simple. We achieve our result by using our network that is resilient to random errors as a black box, to produce a modified family of graphs on which we can perform a Chernoff-union type analysis. The style of our argument provides further motivation for studying the random corruption model, even if the ultimate goal is to be resilient to adversarial corruptions.

Definition 8.

Graph: *Let G^i_{rand} be iid graphs from the distribution G_{rand} all on the same vertex set for $1 \leq i \leq z \triangleq 103 \log n$. Define $G_{wc} \triangleq \bigcup_{i=1}^{z} G^i_{rand}$.*
Protocol: *We now describe the A to B transmission protocol Π_{wc} over this network. The protocol proceeds in two steps:*
(a) A will invoke protocols Π^i_{rand} which will run on the subgraph G^i_{rand}.
(b) B receives z messages each corresponding to one Π^i_{rand}. B takes the majority of all these messages.

Theorem 7. *For the $n = m2^m$-node network $G_{wc} = (V, E)$ and the protocol Π_{wc} for message transmission on it there exists constants α and β, such that whp:*

1. *The network G_{wc} has degree $O(\log n)$.*
2. *The total work is $\text{polylog}(n)$.*
3. *The node complexity is $\text{polylog}(n)$.*
4. *The round complexity is $\text{polylog}(n)$.*
5. *If a subset of nodes $T \subset V$ is corrupt, where $|T| \leq \alpha n$, there exists a set of nodes $S \subset V$ where $|S| \geq n - |T| - \beta |T| / \log n$ such that every pair of nodes in S can communicate reliably with each other by invoking Π_{wc}.*

It is clear that the degree of the graph is $O(\log n)$ and the node complexity, round complexity and total work are $\text{polylog}(n)$.

We proceed to prove resiliency of the protocol. We will first consider a fixed adversary and perform a union bound over all adversaries in the end. We will say the the i^{th} layer has failed if the conditions in Theorem 6 do not hold for G^i_{rand}. Next, we prove that this happens rarely.

Lemma 9. *With probability at least $1 - \binom{n}{t}^{-1.1}$ at most $1/50$ fraction of the layers fail.*

Proof. By Theorem 6 each layer fails with probability $\leq 2^{-n \log (n/t) / \log n}$. So the probability that $2 \log n$ out of $103 \log n$ fail is at most

$$2^{103 \log n} \left(2^{-t \log (n/t) / \log n}\right)^{2 \log n} = \text{poly}(n)(n/t)^{-2t} \ll \binom{n}{t}^{-1.1}$$

for $t = \omega(1)$. □

Theorem 7. Assume less than $1/50$ layers of the network fail. We disregard all the layers that have (as we will see later they will not affect the majority). We still have at least $100 \log n$ layers. Call this new graph G'. Note that we cannot construct this graph and it is only used in the analysis. Define B_i to be the nodes which belong to the unprivileged set for G^i_{rand}. We call a node "give up" for G' if it belongs to $(1/5)$th of the B_i's. By Theorem 6 there are at most $\beta_3|T|$ nodes which are unprivileged per layer. Due to the permutation randomization of V_i the distribution of the set of give up nodes (in G^i_{rand}) is majorized by the distribution in which each node belongs to the set of give up nodes with probability $\beta_3|T|/n$. By Chernoff bound the probability that a particular node(other than bad nodes) is "give up" is $\leq (O(t/n))^{\log n}$. The probability that $\beta_1 t/\log n$ nodes (other than bad nodes) are "give up" is $\leq (O(t/n))^{(\log n)(\beta_1 t/\log n)} = (O(t/n))^{\beta_1 t}$. Now we will do a union bound over all adversaries $\binom{n}{t}(O(t/n))^{\beta_1 t} \ll 1$ for appropriately large constant β_1. □

6 Constant-degree Networks in the Corporation Model

In this section, we show the abundant existence of constant degree networks that are resilient to corporations comprising a constant fraction of nodes being corrupted in the corporation-log(n) model. In Section 4 we built a constant degree graph, G_{rand}, that is resilient to a constant fraction of random corruptions. $G_{rand} = (V = [n], E)$ was constructed by partitioning the nodes of V into n/s committees $C_1, \ldots, C_{n/s}$ (where $s = \log \log n$). Notably, an arbitrary partitioning $\{C_i\}_{i=1}^{n/s}$ sufficed to be resilient to random corruptions. Corporation corruptions are non-random, and in fact adversarial. We show however, the surprising result that essentially the same graph G_{rand} will achieve resilience to corporation-$\log n$ corruptions—even when a constant fraction of the nodes are being corrupted. The result is achieved by inserting more randomness into the structure, leveraging the extremely high probability of success in the G_{rand} construction, and applying—once again—the probabilistic method. To our knowledge, we are the first to introduce the probabilistic method to the field of AE Byzantine fault tolerant networks; and the introduction of this method makes the theorems in this section easy to prove. Surprisingly, we do not know of any methods to achieve these guarantees using previous methods or explicit deterministic constructions. First we present the network and protocol.

Definition 9.

Graph*: The graph $G_{corp} = (V, E)$ is defined almost identically to G_{rand}. The single difference is in how the vertices are partitioned into committees. While in G_{rand} we partition the nodes into* arbitrary *committees of size* $\log \log n$*, in G_{corp} we partition the nodes into* random *committees of the same size.*

Protocol*: The protocol Π_{corp} is identical to the protocol Π_{rand}, just tuned to the new selection of committees.*

Remark 2. By definition, G_{corp} is a random variable, since the breakdown of commitees is defined randomly. However, when it is clear from context, we may refer to a particular realization of the graph as G_{corp} also.

We will refer the to the committees of G_{corp} as $\{C_i\}_{i=1}^{n/s}$ where $s = \log\log n$ is the number of committees.

Lemma 10. *For sufficiently large n, there is a constant ε such that, if any fixed set of $t \leq \varepsilon n$ nodes are corrupted before the committees $\{C_i\}_{i=1}^{n/s}$ are picked at random, then G_{corp} is resilient to these corruptions with probability $1 - 2^{-2t\log(n/t)/\log n}$.*

Proof. Since the corrupted nodes are chosen before the randomization, we can equivalently think of $t \leq \varepsilon n$ nodes being corrupted at random after fixing G_{corp}. So, the theorem is an immediate consequence of Theorem 6. □

We now think of the nodes $[n]$ of G_{corp} as being partitioned among h corporations, as $V_1 \cup \cdots \cup V_h = [n]$, for some value of h

Theorem 8. *For sufficiently large n, there are constants ε, β such that, $(G_{corp}, \mathit{\Pi}_{corp})$ is resilient to a constant fraction of corporation-*$\log n$ *corruptions:*

1. *The network G_{corp} is of constant degree.*
2. *The total work is* $\mathrm{polylog}(n)$.
3. *The node complexity is* $\mathrm{polylog}(n)$.
4. *The round complexity is* $\mathrm{polylog}(n)$.
5. *If any subset $H \subset [h]$ of corporations is corrupted, i.e., the nodes $T = \cup_{i\in H} V_i$ (for $t = |T| \leq \varepsilon n$) is corrupted, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking $\mathit{\Pi}_{corp}$ with probability $1 - 2^{-t\log(n/t)/\log n}$.*

Proof. The first four properties follow from the corresponding properties for G_{rand} in Theorem 6. So, the third property remains to be proved.

Assuming that all the corporations have size at least $k\log n$, the total number of sets H such that $|\cup_{i\in H} V_i| \leq \varepsilon n$ is at most

$$\ell = \sum_{j=0}^{\frac{t}{k\log(n)}} \binom{\frac{n}{k\log(n)}}{j}$$

since the number of indices h is at most $n/(k\log n)$ and at most $\frac{t}{k\log(n)}$ corporations can be corrupted if the total number of nodes corrupted is at most εn. We notice that, $\ell \leq \left(\frac{(e+1)n}{t}\right)^{\frac{t}{k\log n}}$ by Stirling's approximation. The previous lemma shows that the probability of correctness of G_{corp} for t corruptions is $1 - 2^{-2t\log(n/t)/\log n}$. So, using the union bound, we see that making k sufficiently large ensures that the probability of G_{corp} being resilient to t corruptions for all possible adversarial corruptions is at least $1 - 2^{-t\log(n/t)/\log n}$. □

Since, the random network G_{corp} has the properties of the preceding theorem with high probability, it follows by the probabilistic method that there is a fixed graph that has all the properties in the theorem. We distill this straightforward consequence into a concrete corollary.

Corollary 2. *For sufficiently large n, there are constants ε, β and a networked protocol (G, Π) that is resilient to a constant fraction of corporation-$\log n$ corruptions:*

1. *The network G is of constant degree.*
2. *The total work is $\mathrm{polylog}(n)$.*
3. *The node complexity is $\mathrm{polylog}(n)$.*
4. *The round complexity is $\mathrm{polylog}(n)$.*
5. *If any subset $H \subset [h]$ of corporations is corrupted, i.e., the nodes $T = \cup_{i \in H} V_i$ (for $t = |T| \leq \varepsilon n$) is corrupted, there exists a set of nodes $S \subset V$ where $|S| \geq n - \beta|T|$ such that every pair of nodes in S can communicate reliably with each other by invoking Π.*

References

1. Barak, B., Canetti, R., Lindell, Y., Pass, R., Rabin, T.: Secure computation without authentication. In: Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings. pp. 361–377 (2005)
2. Beaver, D., Micali, S., Rogaway, P.: The round complexity of secure protocols (extended abstract). In: Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA. pp. 503–513 (1990)
3. Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In: Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA. pp. 1–10 (1988)
4. Ben-Or, M., Ron, D.: Agreement in the presence of faults, on networks of bounded degree. Inf. Process. Lett. **57**(6), 329–334 (1996)
5. Berman, P., Garay, J.A.: Asymptotically optimal distributed consensus (extended abstract). In: Automata, Languages and Programming, 16th International Colloquium, ICALP89, Stresa, Italy, July 11-15, 1989, Proceedings. pp. 80–94 (1989)
6. Berman, P., Garay, J.A.: Fast consensus in networks of bounded degree (extended abstract). In: Distributed Algorithms, 4th International Workshop, WDAG '90, Bari, Italy, September 24-26, 1990, Proceedings. pp. 321–333 (1990)
7. Boyle, E., Goldwasser, S., Tessaro, S.: Communication locality in secure multiparty computation - how to run sublinear algorithms in a distributed setting. In: Theory of Cryptography - 10th Theory of Cryptography Conference, TCC 2013, Tokyo, Japan, March 3-6, 2013. Proceedings. pp. 356–376 (2013)
8. Canetti, R.: Security and composition of cryptographic protocols: a tutorial (part I). SIGACT News **37**(3), 67–92 (2006)
9. Chandran, N., Chongchitmate, W., Garay, J.A., Goldwasser, S., Ostrovsky, R., Zikas, V.: The hidden graph model: Communication locality and optimal resiliency with adaptive faults. In: Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science, ITCS 2015, Rehovot, Israel, January 11-13, 2015. pp. 153–162 (2015)
10. Chandran, N., Garay, J.A., Ostrovsky, R.: Improved fault tolerance and secure computation on sparse networks. In: Automata, Languages and Programming, 37th International Colloquium, ICALP 2010, Bordeaux, France, July 6-10, 2010, Proceedings, Part II. pp. 249–260 (2010)
11. Chandran, N., Garay, J.A., Ostrovsky, R.: Edge fault tolerance on sparse networks. In: Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part II. pp. 452–463 (2012)
12. Chaum, D., Crépeau, C., Damgård, I.: Multiparty unconditionally secure protocols (extended abstract). In: Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA. pp. 11–19 (1988)
13. Dani, V., King, V., Movahedi, M., Saia, J.: Brief announcement: breaking the o(nm) bit barrier, secure multiparty computation with a static adversary. In: ACM Symposium on Principles of Distributed Computing, PODC '12, Funchal, Madeira, Portugal, July 16-18, 2012. pp. 227–228 (2012)
14. Dolev, D., Fischer, M.J., Fowler, R.J., Lynch, N.A., Strong, H.R.: An efficient algorithm for byzantine agreement without authentication. Information and Control **52**(3), 257–274 (1982)

15. Dolev, D., Reischuk, R.: Bounds on information exchange for byzantine agreement. In: ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, Ottawa, CanadaAugust 18-20, 1982. pp. 132–140 (1982)
16. Dwork, C., Peleg, D., Pippenger, N., Upfal, E.: Fault tolerance in networks of bounded degree (preliminary version). In: Proceedings of the 18th Annual ACM Symposium on Theory of Computing, May 28-30, 1986, Berkeley, California, USA. pp. 370–379 (1986)
17. Garay, J.A., Ostrovsky, R.: Almost-everywhere secure computation. In: Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings. pp. 307–323 (2008)
18. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game or A completeness theorem for protocols with honest majority. In: Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA. pp. 218–229 (1987)
19. King, V., Saia, J.: From almost everywhere to everywhere: Byzantine agreement with $\tilde{o}(n^{3/2})$ bits. In: Distributed Computing, 23rd International Symposium, DISC 2009, Elche, Spain, September 23-25, 2009. Proceedings. pp. 464–478 (2009)
20. King, V., Saia, J.: Breaking the $O(n^2)$ bit barrier: scalable byzantine agreement with an adaptive adversary. In: Proceedings of the 29th Annual ACM Symposium on Principles of Distributed Computing, PODC 2010, Zurich, Switzerland, July 25-28, 2010. pp. 420–429 (2010)
21. King, V., Saia, J., Sanwalani, V., Vee, E.: Scalable leader election. In: Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2006, Miami, Florida, USA, January 22-26, 2006. pp. 990–999 (2006)
22. King, V., Saia, J., Sanwalani, V., Vee, E.: Towards secure and scalable computation in peer-to-peer networks. In: 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), 21-24 October 2006, Berkeley, California, USA, Proceedings. pp. 87–98 (2006)
23. Lamport, L., Shostak, R.E., Pease, M.C.: The byzantine generals problem. ACM Trans. Program. Lang. Syst. **4**(3), 382–401 (1982)
24. Lubotzky, A., Phillips, R., Sarnak, P.: Explicit expanders and the ramanujan conjectures. In: Proceedings of the 18th Annual ACM Symposium on Theory of Computing, May 28-30, 1986, Berkeley, California, USA. pp. 240–246 (1986)
25. Pease, M.C., Shostak, R.E., Lamport, L.: Reaching agreement in the presence of faults. J. ACM **27**(2), 228–234 (1980)
26. Pippenger, N.: On networks of noisy gates. In: 26th Annual Symposium on Foundations of Computer Science, Portland, Oregon, USA, 21-23 October 1985. pp. 30–38 (1985)
27. Upfal, E.: Tolerating linear number of faults in networks of bounded degree. In: Proceedings of the Eleventh Annual ACM Symposium on Principles of Distributed Computing, Vancouver, British Columbia, Canada, August 10-12, 1992. pp. 83–89 (1992)
28. Yao, A.C.: Protocols for secure computations (extended abstract). In: 23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982. pp. 160–164 (1982)