The Hidden Graph Model: Communication Locality and Optimal Resiliency with Adaptive Faults*

Nishanth Chandran
Microsoft Research, India
nichandr@microsoft.com

Wutichai Chongchitmate
UCLA
wutichai@math.ucla.edu

Juan A. Garay
Yahoo Labs
garay@yahoo-inc.com

Shafi Goldwasser†
MIT and The Weizmann Institute of Science
shafi@theory.csail.mit.edu

Rafail Ostrovsky‡
UCLA
rafail@cs.ucla.edu

Vassilis Zikas§
ETH Zurich, Switzerland
vzikas@inf.ethz.ch

ABSTRACT

The vast majority of works on secure multi-party computation (MPC) assume a full communication pattern: every party exchanges messages with *all* the network participants over a complete network of point-to-point channels. This can be problematic in modern large scale networks, where the number of parties can be of the order of millions, as for example when computing on large distributed data.

Motivated by the above observation, Boyle, Goldwasser, and Tessaro [TCC 2013] recently put forward the notion of *communication locality*, namely, the total number of point-to-point channels that each party uses in the protocol, as a quality metric of MPC protocols. They proved that assuming a public-key infrastructure (PKI) and a common reference string (CRS), an MPC protocol can be constructed for computing any n-party function, with communication locality $\mathcal{O}(\log^c n)$ and round complexity $\mathcal{O}(\log^{c'} n)$, for appropriate constants c and c'. Their protocol tolerates a static (i.e., non-adaptive) adversary corrupting up to $t < (\frac{1}{3} - \epsilon)n$ parties for any given constant $0 < \epsilon < \frac{1}{3}$. These results leave open the following questions:
(1) Can we achieve low communication locality and round complexity while tolerating *adaptive* adversaries?
(2) Can we achieve low communication locality with *optimal resiliency* $t < n/2$?

In this work we answer both questions affirmatively. We consider the Boyle *et al.* model, where we replace the CRS with a symmetric-key infrastructure (SKI). In this model we give a protocol with communication locality and round complexity $\mathsf{polylog}(n)$ (similarly to Boyle *et al.*) which tolerates up to $t < n/2$ *adaptive* corruptions, under a standard intractability assumption for adaptively secure protocols, namely, the existence of trapdoor permutations whose domain has invertible sampling. This is done by using the SKI to derive a sequence of random *hidden communication graphs* among players. A central new technique shows how to use these graphs to emulate a complete network in $\mathsf{polylog}(n)$ rounds while preserving $\mathsf{polylog}(n)$ locality. We also show how to remove the SKI setup assumption at the cost, however, of increasing the communication locality (but not the round complexity) by a factor of $\sqrt{n}$.

Categories and Subject Descriptors

F.1.2 [**Modes of Computation**]: Interactive and reactive computation; G.2.2 [**Graph Theory**]: Graph algorithms

General Terms

Theory

1. INTRODUCTION

Secure multi-party computation (MPC for short) allows a set of n parties to securely compute any given function f on their private data. Ensuing the seminal works in the area [40, 26, 2, 14], the systematic study of the problem over the last decades has lead to great improvements regarding several efficiency measures, such as communication complexity (number of exchanged messages), round complexity, and computation complexity. Until recently, however, essentially all MPC results required all parties to communicate directly with each other over a complete network of point to point channels, or by having access to a broadcast channel. While

*The full version of this paper is available at the Cryptology ePrint Archive, `http://eprint.iacr.org/2014/615` .

†Supported in part by NSF Eager CNS-1347364, NSF Frontier CNS-1413920 and Air Force FA8750-11-2-0225.

‡Supported in part by NSF grants CCF-0916574; IIS-1065276; CCF-1016540; CNS-1118126; CNS-1136174; US-Israel BSF grant 2008411, OKAWA Foundation Research Award, IBM Faculty Research Award, Xerox Faculty Research Award, B. John Garrick Foundation Award, Teradata Research Award, and Lockheed-Martin Corporation Research Award. This material is also based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under Contract N00014-11-1-0392. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

§Supported in part by Swiss National Science Foundation (SNF) Ambizione grant PZ00P-2142549.

ITCS'15, January 11–13, 2015, Rehovot, Israel.

http://dx.doi.org/10.1145/2688073.2688102.

this requirement may be harmless when the number of participants is small compared to the complexity of the function f, it is highly problematic in settings where the number of parties is a dominant factor[1].

Communication locality in MPC. Recently, Boyle, Goldwasser, and Tessaro [6], building on work by King *et al.* on Byzantine agreement [31, 32][2], introduced a new efficiency metric called *communication locality* to address such settings. Informally, the communication locality of a protocol is the total number of different point-to-point channels that each party uses in the protocol. The protocols provided in [6] for the computation of any polynomial time function f achieve a communication locality of $\mathsf{polylog}(n)$ assuming a public-key infrastructure (PKI), a common reference string (CRS), and the existence of a semantically secure public-key encryption and existentially unforgeable signatures. An example of a scenario where the complexity of the function may be much smaller than the number of parties, is when securely computing the output of a sublinear algorithm, which takes inputs from a small subset of $q = o(n)$ of parties. (Sublinear algorithms are particularly useful for computing statistics on large populations.) By assuming, in addition to the PKI and semantically secure public-key encryption, the existence of a multi-signature scheme [37, 36], a (certifiable) fully homomorphic encryption (FHE) [7, 8], and simulation-sound adaptive non-interactive zero-knowledge (NIZK) [4, 23], the authors also obtain a protocol for computing sublinear functions, which communicates $\mathcal{O}((\kappa + n) \cdot \mathsf{polylog}(n))$-bit messages[3] and terminates in $\mathsf{polylog}(n) + \mathcal{O}(q)$ rounds.

The solution of [6], however, has the following limitations:

(1) It cannot tolerate an *adaptive* adversary who may choose the parties to corrupt on the fly during the protocol execution; it only tolerates a static adversary who decides on the faulty parties prior to the protocol execution.

(2) It achieves a sub-optimal resiliency of $t < (1/3 - \epsilon)n$ corrupted parties, for any given constant $0 < \epsilon < 1/3$, whereas traditional MPC protocols in the computational setting (without the low communication locality requirement) can tolerate up to $t < n/2$ corruptions.

Our results. In this paper, we first show that by replacing the CRS with a slightly different setup assumption, namely, a *symmetric-key infrastructure (SKI)* [21] where every pair of participants shares a uniformly random key that is unknown to other participants, we can overcome both of the above limitations. Specificially, we construct *adaptively secure* MPC protocols with communication locality $\mathsf{polylog}(n)$ tolerating any $t < n/2$ corruptions. (As mentioned above, this is the optimal number of corruptions that can be tolerated, even in the complete communication setting without the extra requirement of communication locality [26, 15].) Looking ahead, we will show how the SKI can be interpreted as a special type of random initial communication graph which dictates which pairs of players can send point-to-point messages to each other to start with. The graph is shared but "hidden:" each player will only know the restricted subset of $\mathsf{polylog}(n)$ players it can send messages to and receive messages from.[4]

Next, we show that we can remove the additional SKI assumption at the cost of increasing the communication locality by a factor of $\sqrt{n}$. Both our constructions assume the existence of a family of trapdoor permutations which has a *reversed domain sampler* [18, 25]. This is the weakest known general assumption which is sufficient for *non-committing encryption* [10, 18], and thus for adaptively secure MPC over non-private channels. Such families are known to exists under standard number-theoretic assumptions such as the hardness of the decisional Diffie-Hellmann problem (DDH) or the RSA assumption [18].

We remark that in order to circumvent the shortcomings in [6] we need to develop new and quite different techniques, as the limitations to sub-optimal resiliency and non-adaptive adversaries seem to be inherent in their approach. This can be seen as follows. In [6], the parties elect n *input committees* $\mathcal{C}_1, \ldots, \mathcal{C}_n$, as well as one "supreme" committee $\mathcal{C}$—all of size $\mathsf{polylog}(n)$—in a way that ensures that (with high probability) at least a 2/3 fraction of the parties in each committee are honest. Each protocol message of party p_i is then secret-shared to committee $\mathcal{C}_i$, which re-shares it to the parties of the supreme committee $\mathcal{C}$. Subsequently, the members of $\mathcal{C}$ compute the output of the given function on the shared inputs and return it to the users (by sharing it to the input committees, which then reconstruct to their associated input parties). All sharings are private and robust so long as the adversary does not corrupt more than 1/3 of a committee members.

Clearly, the above cannot work if the adversary is allowed to adaptively corrupt parties depending on his view of the election process. Such an adversary might choose to corrupt more than a 1/3 fraction of the parties in some committee[5] and thus violate the privacy of the protocol. Furthermore, even for a static adversary, the above approach cannot yield an optimally resilient (i.e., $t < n/2$) protocol, as an adversary who non-adaptively corrupts $\lceil n/2 \rceil - 1$ of the parties has a noticeable probability of corrupting 1/3 (or even 1/2) of the parties in some committee.

Note that under the additional assumptions of FHE and multi-signatures, [6] obtains better communication complexity for computing sublinear algorithms than directly applying our approach. Improving the communication complexity of our protocols is an enthralling direction for future research.

Other related work. Our result should be contrasted with the work of Dani *et al.* [20], which provides MPC in the information-theoretic setting assuming perfectly private communication channels with communication complexity of $O(\sqrt{n})$, but only offers security against a static adversary and $t < n/3$ corruptions. For the problem of Byzantine agreement (BA), King and Saia [30] show how to construct a protocol that is secure against adaptive corruptions, and where the communication complexity of every party is

[1] Interestingly, recent implementation results report remarkable performance of the state-of-the-art solutions for small instances of the problem such as three-party computation [5] or in a lab environment when broadcast is assumed for free (e.g., [3, 35, 16, 17, 19, 29]).

[2] [31, 32] in fact achieve "almost-everywhere" Byzantine agreement [22], which does not guarantee that all honest players will receive an output.

[3] κ is the security parameter.

[4] In fact, one may alternatively state our setup as having the players share an initial hidden random graph, and our result as a reduction from this setup.

[5] Recall that the adversary has a linear corruption "budget" $t < (1/3 - \epsilon)n$ and the committees are of size $\mathsf{polylog}(n)$.

$\tilde{\mathcal{O}}(\sqrt{n})$. This leads to a BA protocol with $\tilde{\mathcal{O}}(\sqrt{n})$ communication locality; their protocol, however, only tolerates $t < (\frac{1}{3} - \epsilon)n$ corruptions (and is specific to agreement).

Another related body of work is on conducting Byzantine agreement and MPC when players are not connected via a point-to-point network but rather via a sparse, public network. This has been studied both in the context of BA [22, 39, 12, 13] and of MPC [24, 31, 32]. These results inevitably only achieve the so called *almost-everywhere* versions of the problems, as the protocols "give up" a number $x = \omega(1)$ of honest parties (and provide no guarantees for them). The interested reader may refer to the full version [11] for a short survey of the corresponding literature.

1.1 Overview of our results and techniques

In this paper we establish the feasibility of secure multiparty computation with low (i.e., $\mathsf{polylog}(n)$) communication locality both for static and for adaptive adversaries corrupting any $t < n/2$ parties. Our constructions assume a PKI and a symmetric-key infrastructure (SKI—see details below). Furthermore, our protocols have $\mathsf{polylog}(n)$ round complexity. In more detail, we show the following:

THEOREM 1. *Assuming a PKI, an SKI, and trapdoor permutations with a reversed domain sampler, there exists an MPC protocol secure against an* adaptive *adversary corrupting up to $t < n/2$ parties and satisfying the following properties with overwhelming probability:*

- *(Polylogarithmic communication locality) Every party communicates with at most $\mathcal{O}(\log^{1+\epsilon} n)$ other parties, for some constant $\epsilon > 0$.*
- *(Polylogarithmic round complexity) The protocol terminates after $\mathcal{O}(\log^{\epsilon'} n)$ rounds, for some constant $\epsilon' > 0$.*

Since we wish to obtain MPC with guaranteed output delivery for all honest players, our bound on $t < \frac{n}{2}$ is optimal.

Next, we show that we can completely get rid of the SKI setup (while still guaranteeing adaptive security) at the cost of increasing the communication locality (but not the round complexity). That is, we show:

THEOREM 2. *Assuming a PKI and trapdoor permutations with a reversed domain sampler, there exists an MPC protocol secure against an* adaptive *adversary corrupting up to $t < n/2$ parties and satisfying the following conditions with overwhelming probability:*

- *Every party communicates with at most $\mathcal{O}(\sqrt{n}\log^{1+\epsilon} n)$ other parties, for some constant $\epsilon > 0$.*
- *The protocol terminates after $\mathcal{O}(\log^{\epsilon'} n)$ rounds for some constant $\epsilon' > 0$.*

In the remainder of this section we summarize our main techniques and provide a high-level overview of our MPC construction. Before we do that, we describe our model in a bit more detail. All parties are connected via a complete network of point-to-point channels. For simplicity, we assume that the channels are secure; however, as we assume a public-key infrastructure (PKI), these channels can be implemented by encryption and authentication [26]. Furthermore, we assume *synchronous* communication, i.e., our protocols proceed in rounds where messages send in any round are delivered by the end of the round. An adversary can adaptively corrupt $t < n/2$ parties and cannot observe whether or not two honest parties communicated. In addition, our construction assumes a *symmetric-key infrastructure* (denoted SKI), where every pair (i, j) of parties shares a uniformly random key $\mathsf{sk}_{i,j} \in \{0,1\}^\kappa$ for some security parameter κ. Note that there does not seem to be a direct way of getting rid of the SKI assumption without increasing the communication locality, as the direct approach of using the PKI for fair exchange would require (at least) a round where every party communicates with all other parties to exchange the pairwise keys keys. Removing the SKI assumption without increasing the locality is an intriguing open problem.

SKI as a hidden graph setup. Central to our results is a novel way of interpreting/transforming a symmetric key-infrastructure into a special type of setup, which we refer to as *hidden-graph setup* (HG).

Let $G = (V, E)$ be an undirected graph, where $V = [n]$ is the vertex set and E is the set of edges in G. In slight abuse of notation, we also use E to denote the adjacency matrix of G, i.e., $E(i, j) = E(j, i) = 1$ if there is an edge in G connecting vertices i and j; otherwise $E(i, j) = E(j, i) = 0$. We let $G(n, p) = (V, E)$ denote the Erdős-Rényi random graph on n vertices where for every $i, j \in V$, $\Pr[(i, j) \in E] = p$. We refer to such a graph as a *p-random graph*.

We say that the parties in $[n]$ hold a *hidden p-random graph setup* (p-HG)[6] if, after sampling $G = G(n, p)$, every party $i \in [n]$ is given his corresponding row $E(i, j)$ for $j \in [n]$ and no other information on E. Note that instead of the naïve encoding which would require n bits (i.e., give each party the full vector corresponding to his row in E), we can simply give each party i a vector $\Gamma(i)$ which includes the parties i communicates with over the bilateral secure channel. Thus if party i communicates with q parties, his p-HG setup will be of size $q \log(n)$.[7]

We now show how such a HG can be efficiently (and locally) computed from a SKI: Recall that in an SKI every pair of parties i and j is given a uniformly random key $\mathsf{sk}_{i,j}$. We use this key as a seed to a pseudo-random function (PRF). Parties i and j will use the PRF (keyed with $\mathsf{sk}_{i,j}$) to (locally) compute the random coins needed to sample (i, j) for the graph G; i.e., i and j will use the output of the PRF as coins in a sampling algorithm which picks a bit b to be 1 with probability p. If $b = 1$, then i and j will communicate with each other directly in the protocol and (i, j) will be an edge in the communication graph G. The security of the PRF ensures that the bit b computed as above is distributed indistinguishably from the output of the sampling algorithm on uniformly random coins. Without loss of generality, we will henceforth assume that the PRF keys that parties share can be used to sample as many random graphs as needed[8].

Our adaptively secure construction will make use of several ($\mathsf{polylog}(n)$-many) independent HG's. A sequence of ℓ-many HG's that is indistinguishable from a sequence of ℓ independent p-HG's can be generated as above, by querying the PRF on distinct (fixed) inputs.

[6] Throughout this paper we only consider $p = \frac{\log^{1+\epsilon}(n)}{n}$ for some $\epsilon > 0$. Whenever ϵ is clear from the context we might omit p and just refer to the setup as a "(hidden) random graph setup."

[7] In our setting $q = \mathsf{polylog}(n)$ with overwhelming probability, thus, our hidden graph setup is also of size $\mathsf{polylog}(n)$.

[8] Note here, that we can eliminate the need for PRFs by increasing the size of the shared secret key.

Overview of our construction. At the heart of our construction lies a protocol for reliable message transmission (RMT) in this communication-constrained setting. Such a protocol allows a sender i to reliably send a message to a receiver j. Note that as we assume a completely connected network, a trivial way of implementing RMT would be for party i to use the point-to-point channel he shares with each $j \in [n]$. However, our goal is to achieve RMT where each party utilizes only a polylogarithmic number of its direct point-to-point channels. Clearly, in such a setting we cannot allow the adversary to know the neighbors of an honest party $i \in [n]$ as this would enable the adversary to "cut-off" (i.e., isolate) party i from the rest of the parties by corrupting all of its neighbors.

This is where the hidden-graph setup comes in handy: Every party will only exchange messages with its neighbors in this hidden graph and ignore all other interfaces.[9] As we show, an adversary who corrupts up to any constant fraction $q < 1$ of parties cannot make the length of the shortest honest path between any two honest parties to be greater than $\log^{\epsilon'}(n)$, for some $\epsilon' > 0$, except with negligible probability. In particular, we show that if G' denotes the graph that is obtained by deleting from G all parties/nodes that such an adversary corrupts, then with overwhelming probability, every two nodes in G' (i.e., every two honest parties) are connected (in G') by a path of length at most $\log^{\epsilon'} n$. Thus, parties can achieve RMT by simply "flooding" the network; i.e., party i will simply send message m, signed under its signing key, to all its neighbors; then, for $\log^{\epsilon'}(n)$ rounds, all parties in every round, will simply forward (the first validly signed) message that they receive to all its neighbors. Since i and j are connected by a path of length $N = \log^{\epsilon'} n$ in G', then after N rounds, j will receive at least one copy of m that is signed under i's signing key and hence will reliably receive the message m. Observe that the above RMT protocol tolerates any constant fraction $q < 1$ of corruptions (i.e., up to $t \leq qn$ corrupted parties) and requires a standard PKI for digital signatures (in addition to the HG). We assume standard digital signatures secure against chosen-plaintext attacks. Further, since the message is guaranteed to reach all honest parties within N rounds, the above RMT protocol can be used to have a message sent to *all* honest parties.[10]

Unfortunately, the above approach only works for a static adversary. The reason is that, while corrupting parties (even adaptively) and learning their setup, does not reveal anything about the hidden graph (other than the neighbors of corrupted parties themselves), the protocol itself might reveal whether or not $(i, j) \in E$ for honest parties $i, j \in [n]$. For example, if an adversarial party i sends a message to another adversarial party j, and j receives this message in 3 rounds, then it must be the case that there exists a path of length 3 between i and j. One might think that we can get around this problem by simply having i encrypt the message under j's public key; this, however, is completely useless in the case when j is corrupted. Another idea might be to have i delay sending its message; however, this too is useless when i is corrupted.[11] As a result, constructing an RMT protocol for the adaptive-corruption case ends up being much more challenging than in the static case.

The high-level idea behind the protocol for the adaptive case is to sample a new Erdős-Rényi random graph $G = G(n, p)$, with $p = \frac{\log^{\epsilon} n}{n}$, at *every round* of the protocol. As long as the total number of rounds of the protocol is polylogarithmic, so will be the total number of point-to-point channels that an honest party uses (since in each round, every honest party might speak to at most $\mathsf{polylog}(n)$—potentially new—neighbors). The intuition for choosing a different HG for each round is that any corruptions made by the adversary before round i are independent of the graph selected in round i and hence this would be equivalent to the static adversary case. However, now proving that honest parties can communicate reliably (and that there exists a path of bounded length between any two honest parties) is delicate, constituting the crux of our technical result.

Having RMT, the next step is to design the MPC protocol. Recall that our goal is a protocol with full security (i.e., including fairness) and optimal resiliency (i.e., tolerating $t < n/2$ corruptions) [15, 26]. One idea to achieve this is as follows: Since we have already established RMT between any two honest parties, we can invoke any known MPC protocol Π secure for $t < n/2$ assuming authenticated channels, over the virtual network induced by RMT. Whenever party i is instructed in Π to send a message m to party j, we invoke RMT for this purpose. This approach would give an MPC protocol tolerating up to $t < n/2$ corruptions, but does not work generically (for any protocol Π) in combination with our simulated communication channels.

To see why, observe that in our adaptively secure protocol, an increase of the round complexity implies the same (asymptotic) increase of the honest parties' communication locality. Indeed, since using our RMT, every party communicates with $\mathcal{O}(\log^c n)$ (potentially new) parties in every round $1 \leq \ell \leq D$, we can only afford to run a protocol that runs in $\log^{c'} n$ number of rounds for some $c' > 0$. Thus, in order for the above idea to work we need an adaptive MPC protocol over point-to-point authenticated channels which terminates in $\mathsf{polylog}(n)$ rounds. Such a protocol can be obtained by taking any constant-round MPC protocol that utilizes a point-to-point network of secure channels and a broadcast channel (e.g., the protocol in [1]), and modifying it as follows: (1) transmission over the point-to-point secure channels are emulated by calls to our RMT protocol where the message is encrypted using non-committing encryption, and (2) calls to the broadcast channel are emulated by a (randomized, authenticated) broadcast protocol which terminates in $\mathsf{polylog}(n)$ rounds (cf. the protocol in [28]).

REMARK 1 (STATIC SECURITY). *Our primary goal in this paper is adaptive security. However, in the static security setting our approach yields a protocol with* $\mathsf{polylog}(n)$ *locality which relies only on semantically secure public-key encryption and existentially unforgeable signatures (as in [6]). The protocol tolerates an optimal number of* $t < n/2$ *cor-*

[9] Note that the adversary might try to send messages to honest parties using all the corrupted parties. However, the honest parties will ignore messages from all parties that are not their neighbors in their hidden graphs.

[10] Note, however, that if the sender is corrupted, there is no guarantee that the message is sent consistently.

[11] Note that we want to use RMT for *every* pair of parties; thus, the adversary might use information on the HG learned in an execution of RMT with a corrupted sender and/or receiver to attack another RMT with honest sender and receiver.

ruptions and assumes a PKI and a (single) hidden graph setup[12] *(instead of the PKI and CRS assumed in [6]).*

Finally, we show (Section 5) how to avoid the SKI assumption, at the expense of an increased communication locality (but not round complexity)—cf. Theorem 2. In a nutshell, the parties will compute a random graph setup by having each party *locally* decide which of his n point-to-point channels he will use; a channel between two (honest) parties $i, j \in [n]$ is then used only if both parties choose it. By adequately setting the probability of the honest parties' decisions, the resulting communication graph will include an Erdős-Rényi graph which will allow us to use our ideas from the SKI-based construction, with a guaranteed $\mathcal{O}(\sqrt{n}\log^{\delta} n)$ communication locality, for some constant $\delta > 0$.

2. MODEL, DEFINITIONS AND BUILDING BLOCKS

As already mentioned earlier, we assume all parties share a public-key infrastructure (PKI) as well as a symmetric-key infrastructure (SKI). In other words, every party has a public-key, secret-key pair (for a digital signature scheme); every party $i \in [n]$ receives party j's public-key (for all $j \in [n]$). In addition, every pair of parties $i, j \in [n]$ share a secret key $\mathsf{sk}_{i,j}$. Parties are connected by a fully connected *synchronous* network; however, in our constructions every party will only communicate with $\mathsf{polylog}(n)$ other parties.

We allow up to $t < \frac{n}{2}$ parties to be *adaptively* corrupted by a *rushing* adversary (meaning that the adversary is allowed to corrupt parties dynamically during the protocol execution and depending on his view, and that the adversary is able to postpone the sending of any given round's messages until after he receives the messages from the honest parties, resp.).

We consider the standard simulation-based notion of security for multiparty protocols via the real/ideal world paradigm. In other words (and informally), we require that for every probabilistic-polynomial time adversary $\mathcal{A}$ (that corrupts t of the parties) in a real-world execution of the protocol, there exists a corresponding PPT adversary $\mathcal{S}$ in the ideal world who can simulate the output of $\mathcal{A}$ given only access to the ideal world where $\mathcal{S}$ only learns the output of the evaluated function. We prove our results for standalone security. We refer the reader to [9] for further details on this notion of security for multiparty computation. Throughout, we assume that $n > \kappa$, the security parameter.

Our constructions rely on the standard intractability assumption for adaptively secure multi-party protocols, namely, the existence of a family of trapdoor permutations with a reversed domain sampler [18, 25]. Informally, these are trapdoor permutations with an extra property that there exists an algorithm (the reversed domain sampler) which given an input and output can reconstruct (sample) the corresponding random bits used by the function. This assumption is sufficient for all the primitives used in this paper, namely: Pseudo-random functions (PRFs) [27], existentially unforgeable signatures (assuming a PKI setup) [27], constant-round non-committing encryption (informally, this is encryption which transforms an authenticated channel into a secure one in the presence of an adaptive adversary [18]), and constant-round adaptively secure MPC over a point-to-point network with (authenticated) broadcast [1] (see below).

[12]Instead of an SKI, a single copy of our hidden graph can be represented as $\mathsf{polylog}(n)$ bits held by each party corresponding to the vector of the indices of its neighbours.

DEFINITION 3 ([38, 33]). *A protocol for parties $\mathcal{P} = P_1, \cdots, P_n$, where a distinguished player (called the dealer) $P^* \in \mathcal{P}$ holds an initial input m, is a* broadcast protocol tolerating t malicious parties *if the following conditions hold for any adversary controlling at most t parties:*

- **Agreement:** *All honest parties output the same v.*
- **Validity:** *If the dealer is honest, then $v = m$.*

Broadcast protocols that assume a public-key infrastructure are usually termed authenticated.

We also make use of the following fact about expected-constant-round broadcast and Byzantine agreement protocols, implicit in [28].

THEOREM 4 ([28]). *Assuming a PKI, there exists a protocol Π_{BC} which achieves broadcast with overwhelming probability against $t < n/2$ adaptive corruptions, running for $\log^{1+c}(n)$ rounds (constant $c > 0$) on a complete network.*

3. RELIABLE COMMUNICATION IN THE LOCALITY MODEL

In this section we prove our results for Reliable Message Transmission (RMT) between every pair of honest parties in our communication-constrained setting, assuming a standard PKI (for digital signatures) as well as an SKI, as defined above. The constructions in this section tolerate any constant fraction of corrupted parties; that is, we only assume that the number of corrupted parties in $t \leq qn$, for constant $q < 1$ (arbitrarily close to 1).

3.1 Static security

We first show an RMT protocol that is secure against static corruptions. This will illustrate some of the ideas that are needed for our adapively secure construction.

Setup phase. Recall that we work in a model in which parties share a public-key as well as a symmetric-key infrastructure. That is, in the setup phase, party i receives a private key sk_i for a signature scheme, and every party j receives the public key vk_i corresponding to sk_i, for all $i \in [n]$. The SKI allows for a hidden p-random graph setup (p-HG), with $p = \frac{\log^{1+\epsilon} n}{n}$ (for appropriately chosen $\epsilon > 0$), as explained above. Note that, because in this section we assume only a single shared hidden graph, it is sufficient (in fact equivalent) that the keys in the SKI are one-bit long.

Construction idea. The hidden graph setup ensures that the adversary does not get to know whether party i communicates with party j, unless he corrupts one of them. We show that given such a p-HG, an adversary who (non-adaptively) corrupts any constant fraction q of the parties cannot isolate any of the honest parties. In fact, we show a much stronger property for the graph G' formed by removing (in the hidden graph) $t = qn$ corrupted nodes; namely, that with overwhelming probability (in n), every pair (i, j) of honest parties is connected by a path of length at most $N = \log^{\epsilon'}(n)$, for some $\epsilon' > 0$ which depends only on ϵ. Note that since parties start with a PKI, we only require that honest parties $i, j \in [n]$ are connected by a path of length $N = \log^{\epsilon'}(n)$, for some $\epsilon' > 0$ in graph G'. Parties

can then achieve RMT by simply "flooding" the network; i.e., party i will simply send message m, signed under its signing key, to all its neighbors. Next, each party in every round simply forwards the (first validly signed) message that it receives to all of its neighbors. A formal description of the non-adaptively secure protocol for a sender i to reliably send a message m to a receiver j, denoted by $\mathsf{RMT}_{i,j}(m)$, is as follows. (Let $\Gamma(i)$ denote party i's neighbors in G.)

Protocol $\mathsf{RMT}_{i,j}(m)$

1. Round 1: Party i sends $(m, \mathtt{sig}_{\mathsf{sk}_i}(m))$ to all nodes in $\Gamma(i)$.
2. For each round $\rho = 2, \ldots, \log^{\epsilon'}(n)$:
   - For every party $k \in [n] \setminus \{i, j\}$: If a message (m, σ), where σ is party i's valid signature on m, was received *for the first time* from some of its neighbors, i.e., some node in $\Gamma(i)$, in the previous round, then party k sends (m, σ) to all its neighbors and halts. (If multiple validly signed pairs were received in that round for the first time, then take the first one in a lexicographic order.)
   - For receiver j: If a message (m, σ), where σ is party i's valid signature on m, is received *for the first time* from some node in $\Gamma(j)$ then output m and halt. (If multiple validly signed pairs are received in that round for the first time, then take the first one in a lexicographic order.)

The security of protocol $\mathsf{RMT}_{i,j}(m)$ (stated in Theorem 7) can be argued as follows: If i and j are connected by a path of length N in G', then after N rounds j will receive at least one copy of m that is signed under i's signing key, and hence will reliably receive the message m. Thus we simply need to argue that the above holds for some $N = \mathsf{polylog}(n)$. To this direction, we first prove the following lemma, which implies RMT between i and j for all honest $i, j \in [n]$.

LEMMA 5. *Let $G = (V, E)$ be a hidden p-random graph, and let $\mathcal{A}$ be an adversary who non-adaptively chooses a set of parties to corrupt and by doing so learns all their neighbors in G. Denote by $U \subseteq V$ the set of corrupted nodes, and by G' the subgraph on $V \setminus U$ resulting from erasing all nodes in U. If for some constant $q < 1$, $|U| \leq qn$ and $p = \frac{d}{n} = \frac{\log^{1+\epsilon} n}{n}$, then, for any constant $0 < k < \frac{1-q}{2}$, G' is an expander graph with edge expansion kd (except with probability negligible in n).*

PROOF. Since each pair of vertices in G' is still connected with probability p independently of U, G' is a random graph $G((1-q)n, p)$. Let $n' = (1-q)n$ and $0 < k < \frac{1-q}{2}$. Then, for each $S \subseteq V' = V \setminus U$, $|S| = r \leq \frac{n'}{2}$, we have

$$e_{G'}(S, \overline{S}) = \sum_{v \in S, v' \in \overline{S}} X_{v,v'},$$

where $X_{v,v'}$ is the indicator whether there exists an edge between v and v'. Then

$$\mathbb{E}[e_{G'}(S, \overline{S})] = \sum_{v \in S, v' \in \overline{S}} \mathbb{E}[X_{v,v'}] = |S||\overline{S}|p = r(n'-r)p.$$

By the Chernoff bound,

$$\Pr[e_{G'}(S, \overline{S}) < kd|S|] \leq e^{-\left(1 - \frac{kn}{n'-r}\right)^2 r(n'-r)p}$$

$$= \left(e^{-\frac{\left(1-\frac{kn}{n'-r}\right)^2 (n'-r)}{2n}}\right)^{rd} = \left(e^{-\frac{\left(\frac{n'-r}{n} - k\right)^2}{2 \cdot \frac{n'-r}{n}}}\right)^{rd}.$$

Since $0 < r < \frac{n'}{2}$, we have

$$\frac{1-q}{2} = \frac{n'}{2n} \leq \frac{n'-r}{n} \leq \frac{n'}{n} = 1 - q < 1.$$

Thus,

$$\frac{\left(\frac{n'-r}{n} - k\right)^2}{2 \cdot \frac{n'-r}{n}} \geq \frac{1}{2} \cdot \left(\frac{1-q}{2} - k\right)^2 = c > 0.$$

For $d = \log^{1+\epsilon} n$, we have

$$\Pr[e_{G'}(S, \overline{S}) < kd|S|] \leq \left(e^{-c}\right)^{rd} = \left(\frac{1}{n^{c' \log^{\epsilon} n}}\right)^r,$$

and by the union bound, the probability that $e_{G'}(S, \overline{S}) < kd|S|$ for some subset S, $|S| \leq |V'|/2$ is bounded by

$$\sum_{r=1}^{\frac{n'}{2}} \sum_{S, |S|=r} \Pr[e_{G'}(S, \overline{S}) < kd|S|] \leq \sum_{r=1}^{\frac{n'}{2}} \binom{n'}{r} \left(\frac{1}{n^{c' \log^{\epsilon} n}}\right)^r$$

$$< \frac{\frac{1}{n^{c' \log^{\epsilon} n - 1}}}{1 - \frac{1}{n^{c' \log^{\epsilon} n - 1}}} = \lambda(n),$$

where $\lambda(n)$ represents a function that is negligible in n. Therefore, G' is an expander with edge expansion kd with overwhelming probability. □

The next corollary follows immediately from Lemma 5, by using the fact that an expander graph as above has poly-logarithmic diameter except with negligible probability. We make use of the following intuitive terminology: for a given graph $G = ([n], E)$ we say that two parties i and j in $[n]$ are *G-connected by an honest path of length ℓ* if there exists a sequence of connected nodes $\mathtt{PATH}(i, j)$ from i to j in G such that for every node $k \in \mathtt{PATH}(i, j)$, node k is honest, and $|\mathtt{PATH}(i, j)| = \ell$.

COROLLARY 6. *Let $\epsilon > 0$, $p = \frac{\log^{1+\epsilon} n}{n}$, and G be a hidden p-random graph. For any adversary who (non-adaptively) corrupts at most $t = qn$ parties, the following holds except with negligible (in n) probability: there exists some $\epsilon' > 0$ which depends only on ϵ such that any two honest parties are G-connected by an honest path of length at most $\log^{\epsilon'}(n)$.*

The security of protocol $\mathsf{RMT}_{i,j}(m)$ follows now easily from the above corollary, as no matter how the (static) adversary chooses the corrupted parties he cannot increase the diameter of the graph defined by the honest parties and the hidden graph setup to more than $\mathsf{polylog}(n)$.

THEOREM 7. *Let $0 < q < 1$, and $T \subset [n]$ be the set of (non-adaptively) corrupted parties, $|T| = t \leq qn$. Assuming a PKI and an SKI, then $\mathsf{RMT}_{i,j}$ is a secure RMT protocol between any two honest nodes $i, j \in [n] \setminus T$ satisfying the following two conditions with overwhelming probability:*

1. *Every party communicates with at most $\mathcal{O}(\log^{1+\epsilon} n)$ other parties;*
2. *the protocol terminates after $\mathcal{O}(\log^{\epsilon'} n)$ rounds, $\epsilon' > 0$.*

PROOF. Since Lemma 5 shows that any message sent by an honest i will reach every honest j within $\mathcal{O}(\log^{\epsilon'}(n))$ rounds, it follows from the unforgeability property of the signature scheme that j will always accept the message sent by honest i. Hence, the above protocol is a secure RMT

protocol. The communication locality of the protocol follows from the degree of $G = G(n,p)$ which is $\mathcal{O}(\log^{1+\epsilon} n)$, except with negligible probability. □

Parallel composition of RMT. In our MPC construction, we will require all nodes to execute their respective RMT protocols in parallel (simultaneously). That is, let $m_{i,j}$ be the message that node i wishes to send to j via the RMT protocol, denoted $\mathsf{RMT}_{i,j}(m_{i,j})$ as above. Now, let $\mathsf{RMT}_{\mathtt{all}}(\mathbf{m})$ denote the protocol executed by all parties when $\mathsf{RMT}_{i,j}(m_{i,j})$ for all $i,j \in [n]$ are executed in parallel. (That is, in round k of $\mathsf{RMT}_{\mathtt{all}}(\mathbf{m})$, all parties execute the k^{th} round of protocol $\mathsf{RMT}_{i,j}(m_{i,j})$, for all $i,j \in [n]$). $\mathsf{RMT}_{\mathtt{all}}(\cdot)$ is composed of n^2 individual RMT protocols. We have the following corollary.

COROLLARY 8. *For all honest $i,j \in [n]$, $\mathsf{RMT}_{\mathtt{all}}(\mathbf{m})$ is a reliable message transmission protocol for sending $m_{i,j}$ from i to j, satisfying the following properties:*

1. *Every party communicates with at most $\mathcal{O}(\log^{1+\epsilon} n)$ other parties in the protocol.*
2. *The protocol terminates in $\mathcal{O}(\log^{\epsilon'} n)$ rounds, $\epsilon' > 0$.*

PROOF. From Lemma 5 we have that any message sent by any honest i will reach every honest j within $\mathcal{O}(\log^{\epsilon'} n)$ rounds. Hence, from this and the unforgeability of the underlying signature scheme, it follows by a standard hybrid argument that every honest j will always accept the message sent by any honest i at the end of $\mathsf{RMT}_{\mathtt{all}}(\mathbf{m})$. Furthermore, note that the protocol's round complexity is equal to the maximum round complexity of its components, which equals $\mathcal{O}(\log^{\epsilon'} n)$. Further, note that the communication locality of every party in $\mathsf{RMT}_{\mathtt{all}}(\mathbf{m})$ is equal to the communication locality of the party in $\mathsf{RMT}_{i,j}(m_{i,j})$, for any $i,j \in [n]$. Hence, the corollary follows. □

3.2 Adaptively secure RMT

As discussed in the Section 1.1 the above proof technique fails against adaptive adversaries. Informally, the issue is that an adversary can use the round in which a corrupted party/relayer receives a message to deduce information on the communication graph (see Section 1.1 for more details and a concrete example). In this section we describe an RMT protocol that is secure against such an *adaptive* adversary. The idea is have the parties use a different, independent communication graph for each round in the transmission scheme. As long as the transmission scheme does not have more than $\mathsf{polylog}(n)$ rounds and in each round, every party communicates with at most $\mathsf{polylog}(n)$ (additional) parties, the overall locality with be $\mathsf{polylog}(n)$.

The main challenge in the above idea is to prove that in this dynamically updated communication graph, the message will reach each recipient through an honest path in at most $\mathsf{polylog}(n)$ rounds. Proving this constitute the main technical contribution of our work. The (adaptively secure) RMT protocol AdRMT is similar to the protocol in the static case, except that in round ρ parties forward messages received in the previous round to their neighbours in the communication graph G_ρ. We first describe the corresponding setup that it requires.

Setup phase. As in the static case, the parties share both a PKI and an SKI. The SKI will be used here in the same spirit, except that instead of generating one Erdős-Rényi graph, $G = G(n,p)$ with $p = \frac{\log^{\epsilon} n}{n}$, it will be used to generate D such graphs, denoted $\mathcal{G} = (G_1, \ldots, G_D)$. These graphs can be sampled using the same PRF key $\mathsf{sk}_{i,j}$ that parties i and j share. As before, every node only knows its own neighbors, and when the adversary corrupts a node j, he only learns j's neighbors in $G_1, \ldots, G_D$.

The protocol is described below, followed by security statement and a high-level description of its proof. (The formal proof can be found in the full version.)

Protocol $\mathsf{AdRMT}_{i,j}(m)$

1. Round 1: Party i sends $(m, \mathtt{sig}_{\mathsf{sk}_i}(m))$ to all its neighbors in graph G_1.
2. For each round $\rho = 2, \ldots, \log^{\epsilon'}(n)$:
   - For every party $k \in [n] \setminus \{i,j\}$: If a message (m,σ), where σ is party i's valid signature on m was received *for the first time* from some of its neighbours in $G_{\rho-1}$ in the previous round, then party k sends (m,σ) to all its neighbors in graph G_ρ and halts. (If multiple validly signed pairs were received in that round for the first time, then take the first one in a lexicographic order.)
   - For receiver j: If a message (m,σ), where σ is party i's valid signature on m is received *for the first time* from some of party j's neighbours in G_ρ, then output m and halt. (If more than one validly signed pair is received in that round for the first time, then take the first one in a lexicographic order.)

THEOREM 9. *Let $T \subset [n]$ be the set of adaptively corrupted parties, $|T| = t \leq qn$, for any constant $0 < q < 1$. Assuming a PKI and an SKI, protocol $\mathsf{AdRMT}_{i,j}(m)$ is a secure RMT protocol between any two honest nodes $i,j \in [n] \setminus T$, satisfying the following two properties with overwhelming probability:*

1. *Every party communicates with at most $\mathcal{O}(\log^{1+\epsilon} n)$ other parties.*
2. *The protocol terminates in $\mathcal{O}(\log^{\epsilon'} n)$ rounds, $\epsilon' > 0$.*

Proof idea. As in the static case, we show that there exists a path of length at most $\mathcal{O}(\log^{\epsilon'}(n))$ between any two honest nodes $i,j \in [n]$ when we consider the collection of communication graphs $\mathcal{G}$ that selects graph G_i as the communication graph in hop i. We prove this in three steps:

First, we prove that at every step of the protocol, even if an adversary corrupts a constant fraction of the nodes in the random graph, the honest neighbors of any set S of size $\leq \frac{n}{d}$ that are not in S, will be at least of size $kd|S|$, for some appropriate constant k (except with negligible probability). More concretely, in the full version, we prove the following lemma, where we let $\epsilon > 0, 0 < q < 1$ be constants, $d = \log^{1+\epsilon} n$, $p = \frac{d}{n} = \frac{\log^{1+\epsilon} n}{n}$, and $D = O(\log n)$.

LEMMA 10. *Let $G = G(n,p)$ be graph on $V = [n]$, and $U \subseteq V$, $|U| \leq qn$, chosen adaptively while only learning edges connecting to U. Let G' be the induced subgraph on $V' = V \setminus U$. Then, for any constant $0 < k < \frac{1-q}{2}$, there exists a constant $c > 0$ such that, for sufficiently large n and for any $S \subseteq V'$ with $|S| = r \leq \frac{n}{d} = \frac{1}{p}$, the set of all neighbors of S that are not in S, $\Gamma(S)$, has size at least $kd|S|$ except with negligible probability $P_r = \left(\frac{1}{n^{c \log^{\epsilon} n}}\right)^r$.*

Next, via an application of Hoeffding's inequality, we prove that as long as the adversarial set of parties (of size at most qn for some constant $0 < q < 1$) are chosen independently of the random neighbors chosen by any party, a constant fraction of the party's neighbors will be honest, except with negligible probability. Thus, we get:

LEMMA 11. *Let $V = [n]$ and $C \subseteq V$, $|C| = m$, be a subset chosen uniformly at random. Let $0 < q < 1$ be a constant and $U \subseteq V$, $|U| = qn$, be a subset chosen independently of C. Then, for all $0 < \delta < 1-q$, $|C \setminus U| > (1-q-\delta)m$ except with probability $e^{-2m\delta^2}$. In particular, for $m = \log^{1+\epsilon'} n$, $|C \setminus U| > \left(\frac{1-q}{2}\right) m$ except with negligible probability. Further, for $q = \frac{1}{2} - \epsilon$, $|C \setminus U| > \frac{1}{2}m$ except with negligible probability.*

Finally, using Lemmas 10 and 11, we show that even when an adaptive adversary corrupts parties in every round of the protocol, if the parties select a random graph at each round of the protocol, there exists a path of length at most $D = \mathcal{O}(\log n)$ between any two honest nodes in $[n]$. Formally:

LEMMA 12. *Let $G_1, \ldots, G_D$ be graphs on $V = [n]$ constructed independently as $G(n,p)$. Let $U_1, U_2, \ldots, U_D \subseteq V$ be disjoint subsets with $U = \cup_{i=j}^{D} U_j$ such that $|U| = qn$ where U_j is chosen independently from $G_{j+1}, \ldots, G_D$, but adaptively, after learning the neighbors of U_i in G_i for $i \leq j$. Let G'_i be the induced subgraph on $V_i = V \setminus (\cup_{j=1}^{i} U_j)$. Then, except with negligible probability, any pair of vertices $v, v' \in V' = V \setminus U$ are reachable with respect to $\mathcal{G}' = (G'_1, \ldots, G'_D)$ by a path of length at most D.*

Combining these gives us our main theorem (Theorem 9). □

Parallel composition of adaptively secure RMT. Once again, we will require all nodes $i, j \in [n]$ to execute their respective RMT protocols in parallel simultaneously. Let $\mathsf{AdRMT}_{\mathtt{all}}(\mathbf{m})$ denote the protocol executed by all parties when $\mathsf{AdRMT}_{i,j}(m_{i,j})$ for all $i, j \in [n]$ are executed in parallel. That is, in round k of $\mathsf{AdRMT}_{\mathtt{all}}(\mathbf{m})$, all parties execute the k^{th} round of protocol $\mathsf{AdRMT}_{i,j}(m_{i,j})$ (for all $i, j \in [n]$). Note that the graph G_k used in the k^{th} round of the protocol depends only on the round k and not on i and j; hence, we use the same graph G_k to send all the messages of protocol $\mathsf{AdRMT}_{\mathtt{all}}(\mathbf{m})$. We have the following corollary:

COROLLARY 13. *For all honest $i, j \in [n]$, $\mathsf{AdRMT}_{\mathtt{all}}(\mathbf{m})$ is a reliable message transmission protocol for sending $m_{i,j}$ from i to j, satisfying the following properties:*

1. *Every party communicates with at most $\mathcal{O}(\log^{1+\epsilon} n)$ other parties in the protocol.*
2. *The protocol terminates in $\mathcal{O}(\log^{\epsilon'} n)$ rounds, $\epsilon' > 0$.*

The proof of this corollary is similar to Corollary 8's.

4. SECURE MULTI-PARTY COMPUTATION WITH LOW COMMUNICATION

We are now ready to describe our MPC protocol for securely evaluating any given (even reactive) n-party function in the comunication-locality model. Our protocol is secure against $t < n/2$ adaptive corruptions. The idea behind our MPC protocol is to use a constant-round adaptively secure MPC protocol for $t < n/2$ working over point-to-point secure channels and broadcast (e.g., [1]), where those resources are emulated via our RMT protocol of Section 3.2.

We let $\Pi_{\mathtt{BC}}$ denote the authenticated broadcast protocol guaranteed by Theorem 4 (Section 2). The protocol achieves broadcast with overwhelming probability against $t < n/2$ adaptive corruptions, running for $\log^{1+c} n$ rounds on a complete network, for some constant $c > 0$. As pointed out in [28], assuming unique process and message ID's as in [34], $\Pi_{\mathtt{BC}}$ remains secure under parallel composition.

Let $\Pi^*_{\mathtt{BC}}$ denote the protocol which results by having the parties execute $\Pi_{\mathtt{BC}}$ where in each round instead of using the point-to-point channels for exchanging their messages, the parties invoke $\mathsf{AdRMT}_{\mathtt{all}}$ from Section 3.2. Then it follows immediately from the security of $\mathsf{AdRMT}_{\mathtt{all}}$ (Corollary 13) and the fact that each message transmission requires $\mathsf{polylog}(n)$ rounds that protocol $\Pi^*_{\mathtt{BC}}$ is also a secure broadcast protocol with polylogarithmic round complexity and communication locality.

LEMMA 14. *Protocol $\Pi^*_{\mathtt{BC}}$ described above achieves broadcast against $t < n/2$ adaptive corruptions and satisfies the following conditions with overwhelming probability:*

1. *Every party communicaties with at most $\mathcal{O}(\log^{1+\epsilon} n)$ parties.*
2. *The protocol terminates in $\mathcal{O}(\log^{\epsilon'} n)$ rounds, $\epsilon' > 0$.*

PROOF. The security of $\Pi^*_{\mathtt{BC}}$ follows directly from the security of protocols $\Pi_{\mathtt{BC}}$ and $\mathsf{AdRMT}_{\mathtt{all}}$. The (asymptotic) round complexity is computed as follows: for each round ℓ of $\Pi_{\mathtt{BC}}$, protocol $\Pi^*_{\mathtt{BC}}$ executes $\mathsf{AdRMT}_{\mathtt{all}}$ to have the parties exchange their round ℓ messages; thus, for each round in $\Pi_{\mathtt{BC}}$ we need $\mathcal{O}(\log^{\epsilon''} n)$ rounds in $\Pi^*_{\mathtt{BC}}$. Because $\Pi_{\mathtt{BC}}$ runs in $\mathcal{O}(\log^{\epsilon'} n)$ rounds, the total round complexity of $\Pi^*_{\mathtt{BC}}$ is $\mathcal{O}(\log^{\epsilon'+\epsilon''} n)$ rounds. We next argue the communication locality: With overwhelming probability, in each round of $\Pi^*_{\mathtt{BC}}$, every party might communicate with at most to $\mathcal{O}(\log^{1+\epsilon} n)$ (potentially different) parties (for executing $\mathsf{AdRMT}_{\mathtt{all}}$). Thus, since the total number of rounds is $\mathcal{O}(\log^{\epsilon'+\epsilon''} n)$, then with overwhelming probability (by the union bound) the total number of parties that each $i \in [n]$ exchanges messages with using the point-to-point channels is $\mathcal{O}(\log^{1+\epsilon+\epsilon'+\epsilon''} n)$. □

The next step is to construct a *secure* message transmission protocol (SMT) which allows a sender i to securely (i.e., authentically and privately) send a message $m_{i,j}$ to a receiver j. Since we have a PKI and an adaptively secure broadcast protocol, we can use the standard reduction of secure channels to broadcast: The sender i encrypts $m_{i,j}$ under the receiver's public key and broadcasts the corresponding ciphertext $c_{i,j}$. Upon receiving $c_{i,j}$, party j decrypts it using his secret key and recovers $m_{i,j}$. However, for the above reduction to be secure (in a simulation-based manner) against an adaptive adversary, we must ensure that a simulator can "open" a ciphertext to any message of its choice. This can be achieved by the use of a *non-committing encryption* for computing $c_{i,j}$ [10]. As proved in [18] constant-round non-committing encryption can be constructed assuming the existence of families of trapdoor permutations with a reversed domain sampler. We use $\mathsf{AdSMT}_{i,j}$ to denote the above SMT protocol, and $\mathsf{AdSMT}_{\mathtt{all}}$ to denote the protocol composed of n^2 individual $\mathsf{AdSMT}_{i,j}(m_{i,j})$ protocols (for all $i, j \in [n]$), run in parallel, where $\mathbf{m} = (m_{1,1}, m_{1,2}, \ldots, m_{nn})$. We now prove Theorem 1.

PROOF. Let $\Pi_{\mathtt{MPC}}$ denote a constant-round MPC protocol which is secure against adaptive corruptions of up to

$t < n/2$ parties, where parties communicate over a complete network of point-to-point channels and broadcast. (Such protocols are known to exist under the assumption in the theorem, e.g., [1].) Furthermore, let $\Pi^*_{\texttt{MPC}}$ denote the protocol that results by instantiating in $\Pi_{\texttt{MPC}}$ the calls to the secure channels and broadcast by invocations of protocols $\Pi^*_{\texttt{BC}}$ and AdSMT, respectively. We argue that $\Pi^*_{\texttt{MPC}}$ satisfies all the properties claimed in the theorem. The security of $\Pi^*_{\texttt{MPC}}$ follows immediately from the security of the underlying protocol $\Pi_{\texttt{MPC}}$ and the security of protocols $\Pi^*_{\texttt{BC}}$ and $\mathsf{AdSMT}_{\texttt{all}}$. For the round complexity: For each round in $\Pi_{\texttt{MPC}}$, all message exchanges (i.e., point-to-point transmissions or broadcast calls) are exchanged in $\Pi^*_{\texttt{MPC}}$ by appropriate (parallel) executions of protocols $\Pi^*_{\texttt{BC}}$ and $\mathsf{AdSMT}_{\texttt{all}}$, where the executions have unique round, protocol, and message IDs.[13] Thus, for every round in $\Pi_{\texttt{MPC}}$ we need $\mathcal{O}(\log^{\epsilon'} n)$ rounds in $\Pi^*_{\texttt{MPC}}$, for some given constant $\epsilon' > 0$. Because $\Pi_{\texttt{MPC}}$ terminates in a constant number of rounds, the round complexity of $\Pi^*_{\texttt{MPC}}$ is also $\mathcal{O}(\log^{\epsilon'} n)$. In each of these rounds, every party might communicate with at most $\mathcal{O}(\log^{1+\epsilon} n)$ (potentially different) parties, (Recall that all parallel executions of $\Pi^*_{\texttt{BC}}$ and $\mathsf{AdSMT}_{\texttt{all}}$ use the same sequence of graph setups.) Thus, the total number of parties that each $i \in [n]$ talks directly to (i.e., via its point-to-point channels) is $\mathcal{O}(\log^{1+\epsilon+\epsilon'} n)$. □

5. GETTING RID OF THE SKI

In this section we show how to get rid of the symmetric-key setup assumption, at the cost, however, of increasing the communication-locality (but not the round complexity) by a factor of $\sqrt{n}$. The idea for getting rid of the SKI is to have the parties compute some kind of an alternative random graph setup. This is done as follows: each party $i \in [n]$ locally decides which of his n point-to-point channels he will use; a channel between two (honest) parties $i, j \in [n]$ is then used only if both parties choose it. (This is similar in spirit to the way the work of Chandran *et al.* [13] handles "edge corruptions" in sparse networks.) By having each party decide to use each of his channels with probability $p = \frac{\log^{\epsilon} n}{\sqrt{n}}$ for some given constant $\epsilon > 1$ (and ignore all other channels) we ensure that, with overwhelming probability, each (honest) party uses at most $\mathcal{O}(\sqrt{n}\log^{\delta} n)$ of its point-to-point channels for some constant $\delta > 0$. Furthermore, each edge between two honest parties i and j is chosen with probability $p' = p^2 = \frac{\log^{2\epsilon} n}{n}$, thus the resulting communication graph will include Erdős-Rényi graph $G(n, p')$ which will allow us to use our ideas from the previous sections. Note, that as the adversarial nodes might choose to communicate with all their neighbors, the communication locality is no longer guaranteed to be $\mathcal{O}(\log^{\epsilon} n)$; notwithstanding, it is guaranteed to be $\mathcal{O}(\sqrt{n}\log^{\delta} n)$ with overwhelming probability.

RMT protocol. We now describe a reliable message transmission protocol that tolerates up to $t < qn$ adaptive corruptions, for any constant $q < 1$. Our protocol is similar to the corresponding protocol from Section 3.2, the only difference being that the parties choose their neighbors in a setup procedure as above instead of sampling them through their SKI-keys. We then obtain the following theorem: the proof of communication locality follows from an application of the Chernoff bound and the proof of round complexity is similar to the proof of Theorem 9 (see the full version for details).

[13]Recall that the ID's are needed to ensure security of $\Pi^*_{\texttt{BC}}$ under parallel composition [34].

Protocol $\mathsf{AdRMT}^{\texttt{noSKI}}_{i,j}(m)$

1. Round 1 (Computing the setup): The parties execute the following code for every $(i, j, \rho) \in [n] \times [n] \times [\log^{\epsilon'} n]$ *in parallel* (where $\epsilon' > 1$ is a given constant):
   - Party i samples a bit $b^{\rho}_{i,j}$ where $b^{\rho}_{i,j} = 1$ with probability $p = \frac{\log^{\epsilon} n}{\sqrt{n}}$ for some given constant $\epsilon > 1$; and $b^{\rho}_{i,j} = 0$ otherwise.
   - If $b^{\rho}_{i,j} = 0$ for all $\rho \in [\log^{\epsilon'} n]$, then party i ignores all messages on the point-to-point channel between i and j.
   - If $b^{\rho}_{i,j} = 1$ then i sends $(b^{\rho}_{i,j}, \rho)$ to j.
2. Round 2: For each $(i, j, \rho) \in [n] \times [n] \times [\log^{\epsilon'} n]$: If $b^{\rho}_{i,j} = 1$ but party i received no message (b, ρ) from party j in the previous round then i sets $b^{\rho}_{i,j} := 0$. For $\rho = 1, \ldots, \log^{\epsilon'} n$: Party i sets $\Gamma(i)^{\rho} := \{j \mid b^{\rho}_{i,j} = 1\}$ to be the set of parties/neighbors p_i will communicate with in round ρ.
3. Round 3: Party i sends $(m, \mathtt{sig}_{\mathtt{sk}_i}(m))$ to parties in $\Gamma(i)^{\rho}$.
4. For each round $\rho = 3, \ldots, \log^{\epsilon'} n$:
   - For every party $k \in [n] \setminus \{i, j\}$: If a message (m, σ), where σ is party i's valid signature on m was received *for the first time* in the previous round $\rho - 1$ from some party in $\Gamma(k)^{\rho-1}$, then party k sends (m, σ) to all parties in $\Gamma(k)^{\rho}$ and halts. (If multiple validly signed pairs were received in that round for the first time, then take the first one in a lexicographic order.)
   - For the receiver j: If a message (m, σ), where σ is party i's valid signature on m is received *for the first time* from some party in $\Gamma(j)^{\rho}$, then output m and halt. (If more than one validly signed pair is received in that round for the first time, then take the first one in a lexicographic order.)

THEOREM 15. *Let $T \subset [n]$ be the set of adaptively corrupted parties, $|T| = t \leq qn$, for any constant $0 < q < 1$. Assuming a PKI, protocol $\mathsf{AdRMT}^{\texttt{noSKI}}_{i,j}(m)$ is a secure RMT protocol between any two honest nodes $i, j \in [n]\backslash T$, satisfying the following tow properties with overwhelming probability:*

1. *Every party communicates with at most $\mathcal{O}(\sqrt{n}\log^{1+\delta} n)$ other parties, for some constant $\delta > 0$.*
2. *The protocol terminates in $\mathcal{O}(\log^{\epsilon''} n)$ rounds, for some constant $\epsilon'' > 0$.*

Given Theorem 15, an MPC protocol with the desired communication locality and round complexity can be obtained by replacing in protocol $\Pi^*_{\texttt{MPC}}$ all invokations of $\mathsf{AdRMT}_{i,j}$ with invocations of $\mathsf{AdRMT}^{\texttt{noSKI}}_{i,j}$. The proof of Theorem 2 is similar to the proof of Theorem 1.

6. REFERENCES

[1] D. Beaver, S. Micali, and P. Rogaway. The round complexity of secure protocols (extended abstract). In *STOC*, pages 503–513, 1990.

[2] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *STOC*, pages 1–10, 1988.

[3] R. Bendlin, I. Damgård, C. Orlandi, and S. Zakarias. Semi-homomorphic encryption and multiparty computation. In *EUROCRYPT*, pages 169–188, 2011.

[4] M. Blum, P. Feldman, and S. Micali. Non-interactive zero-knowledge and its applications (extended abstract). In *STOC*, pages 103–112, 1988.

[5] D. Bogdanov, S. Laur, and J. Willemson. Sharemind: A framework for fast privacy-preserving computations. In *ESORICS*, pages 192–206, 2008.

[6] E. Boyle, S. Goldwasser, and S. Tessaro. Communication locality in secure multi-party computation - how to run sublinear algorithms in a distributed setting. In *TCC*, pages 356–376, 2013.

[7] Z. Brakerski, C. Gentry, and V. Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In *ITCS*, pages 309–325, 2012.

[8] Z. Brakerski and V. Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In *FOCS*, pages 97–106, 2011.

[9] R. Canetti. Security and composition of cryptographic protocols: a tutorial (part i). *SIGACT News*, 37(3):67–92, 2006.

[10] R. Canetti, U. Feige, O. Goldreich, and M. Naor. Adaptively secure multi-party computation. In *STOC*, pages 639–648, 1996.

[11] N. Chandran, W. Chongchitmate, J. A. Garay, S. Goldwasser, R. Ostrovsky, and V. Zikas. Optimally resilient and adaptively secure multi-party computation with low communication locality. Cryptology ePrint Archive, Report 2014/615, 2014.

[12] N. Chandran, J. A. Garay, and R. Ostrovsky. Improved fault tolerance and secure computation on sparse networks. In *ICALP*, pages 249–260, 2010.

[13] N. Chandran, J. A. Garay, and R. Ostrovsky. Edge fault tolerance on sparse networks. In *ICALP*, pages 452–463, 2012.

[14] D. Chaum, C. Crépeau, and I. Damgård. Multiparty unconditionally secure protocols (extended abstract). In *STOC*, pages 11–19, 1988.

[15] R. Cleve. Limits on the security of coin flips when half the processors are faulty (extended abstract). In *STOC*, pages 364–369, 1986.

[16] I. Damgård, M. Keller, E. Larraia, C. Miles, and N. Smart. Implementing AES via an actively/covertly secure dishonest-majority MPC protocol. In *SCN*, pages 241–263, 2012.

[17] I. Damgård, M. Keller, E. Larraia, V. Pastro, P. Scholl, and N. Smart. Practical covertly secure MPC for dishonest majority - or: Breaking the SPDZ limits. In *ESORICS*, pages 1–18, 2013.

[18] I. Damgård and J.B. Nielsen. Improved non-committing encryption schemes based on a general complexity assumption. In *CRYPTO*, pages 432–450, 2000.

[19] I. Damgård, V. Pastro, N. Smart, and S. Zakarias. Multiparty computation from somewhat homomorphic encryption. In *CRYPTO*, pages 643–662, 2012.

[20] V. Dani, V. King, M. Movahedi, and J. Saia. Brief announcement: breaking the o(nm) bit barrier, secure multiparty computation with a static adversary. In *PODC*, pages 227–228, 2012.

[21] Y. Dodis, J. Katz, A. Smith, and S. Walfish. Composability and on-line deniability of authentication. In *TCC*, pages 146–162, 2009.

[22] C. Dwork, D. Peleg, N. Pippenger, and E. Upfal. Fault tolerance in networks of bounded degree (preliminary version). In *STOC*, pages 370–379, 1986.

[23] U. Feige, D. Lapidot, and A. Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *FOCS*, pages 308–317, 1990.

[24] J. A. Garay and R. Ostrovsky. Almost-everywhere secure computation. In *EUROCRYPT*, pages 307–323, 2008.

[25] O. Goldreich. Basing non-interactive zero-knowledge on (enhanced) trapdoor permutations: The state of the art. In *Studies in Complexity and Cryptography*, pages 406–421. Springer-Verlag, 2011.

[26] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In *STOC*, pages 218–229, 1987.

[27] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *FOCS*, pages 230–235, 1989.

[28] J. Katz and C. Y. Koo. On expected constant-round protocols for byzantine agreement. In *CRYPTO*, pages 445–462, 2006.

[29] M. Keller, P. Scholl, and N. Smart. An architecture for practical actively secure MPC with dishonest majority. In *CCS*, pages 549–560, 2013.

[30] V. King and J. Saia. Breaking the $O(n^2)$ bit barrier: scalable byzantine agreement with an adaptive adversary. In *PODC*, pages 420–429, 2010.

[31] V. King, J. Saia, V. Sanwalani, and E. Vee. Scalable leader election. In *SODA*, pages 990–999, 2006.

[32] V. King, J. Saia, V. Sanwalani, and E. Vee. Towards secure and scalable computation in peer-to-peer networks. In *FOCS*, pages 87–98, 2006.

[33] L. Lamport, R. E. Shostak, and M. C. Pease. The byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(3):382–401, 1982.

[34] Y. Lindell, A. Lysyanskaya, and T. Rabin. On the composition of authenticated byzantine agreement. In *STOC*, pages 514–523, 2002.

[35] Y. Lindell, E. Oxman, and B. Pinkas. The IPS compiler: Optimizations, variants and concrete efficiency. In *CRYPTO*, pages 259–276, 2011.

[36] S. Lu, R. Ostrovsky, A. Sahai, H. Shacham, and B. Waters. Sequential aggregate signatures and multisignatures without random oracles. In *EUROCRYPT*, pages 465–485, 2006.

[37] S. Micali, K. Ohta, and L. Reyzin. Accountable-subgroup multisignatures: extended abstract. In *CCS*, pages 245–254, 2001.

[38] M. C. Pease, R. E. Shostak, and L. Lamport. Reaching agreement in the presence of faults. *J. ACM*, 27(2):228–234, 1980.

[39] E. Upfal. Tolerating linear number of faults in networks of bounded degree. In *PODC*, pages 83–89, 1992.

[40] A. C. Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pages 162–167, 1986.