Secure Communication in Minimal Connectivity Models

Matthew Franklin and Rebecca N. Wright

AT&T Labs Research, 180 Park Avenue, Florham Park, NJ 07932, USA

Abstract. Problems of secure communication and computation have been studied extensively in network models. In this work, we ask what is possible in the information-theoretic setting when the adversary is very strong (Byzantine) and the network connectivity is very low (minimum needed for crash-tolerance). For some natural models, our results imply a sizable gap between the connectivity required for *perfect* security and for *probabilistic* security. Our results also have implications to the commonly studied simple channel model and to general secure multiparty computation.

1 Introduction

If two parties are connected by a private and authenticated channel, then secure communication between them is guaranteed. However, in most networks, many parties are only indirectly connected, as elements of an incomplete network of private and authenticated channels. The interplay of network connectivity and secure communication has received a lot of attention in the literature [Dol82,BGW88,CCD88,Bea89,RB89,BCG93,DDWY93]. Not only is secure communication important in its own right, but it is also an essential primitive from which general secure computation can be achieved [BGW88,CCD88].

Much is known if the channels are *simple*, i.e., if each channel has a single sender and a single recipient. If there are t faulty processors, and the faults are passive *gossipers*, then $t+1$ disjoint paths of channels between sender and receiver are necessary and sufficient for secure communication. The same is true for a setting in which the only faults are crash failures. If the t faulty processors are active *Byzantine* failures, under the control of a computationally unbounded adversary, then $2t+1$ disjoint paths between sender and receiver are necessary and sufficient [DDWY93]. Notice the gap in the connectivity required to tolerate a weak adversary and a strong one.

Less is known when a channel may have multiple recipients. The case of passive faults in multi-recipient networks has been studied previously [FY95]. The case of active faults in the *public broadcast* model (which can be thought of as the largest possible multi-recipient channels) has also been studied previously [GGL91]. In this paper, we begin the study of active faults for multi-recipient channels.

Following Dolev et al. [DDWY93], we abstract away the network and consider that sender and recipient are connected by some number of *wires* or *simple*

Appears in Proceedings of Eurocrypt '98.

lines. Each wire is a disjoint collection of processors arranged linearly, with communication links only between adjacent processors. We add the assumption that anything sent to a neighbor on any line is received identically by the other neighbor, whether or not the originator is faulty. In the literature, this is known as *reliable multicast* [PSL80,Ch85,PG89]. Hence, we call this property *multicast*, turning simple lines into *multicast lines*.

Note however that, unlike in the simple channel model, it is not possible to directly apply protocols over multicast lines to disjoint paths in a general multicast graph, because disjoint paths may have common neighbors. We focus on multicast lines as a first step towards understanding the power of the multicast model. It is not immediately obvious whether the change to multiple receivers helps or hurts an active adversary. On one hand, the adversary may benefit from the loss in privacy of every channel. On the other hand, the adversary too suffers from a restriction, since an incorrect transmission from a faulty processor on a channel will always be received identically by all of that channel's receivers. As we will see, this change hurts the adversary more than it helps.

Our model is related to that of Bracha and Toueg [BT85], who use *echo-broadcast* to refer to a primitive that restricts the communication behavior of a faulty processor so that contradictory messages are not received by different parties. We remark that the radio network model studied by Alon et al. [ABLP89] is somewhat different from what we consider here. Their work addresses issues of coordination and scheduling that arise in packet radio networks, and does not consider privacy. Note that it is implicit in our model that all nodes know the full network structure. Burmester et al. [BDK97] show that the situation may be quite different if the network structure is not known by all parties.

Our Results: This paper has two main areas of contribution. First, we provide a complete characterization of when secure communication is possible over multicast lines and an almost complete characterization of when it is efficient. Second, we compare the power of multicast lines to the power of simple lines alone and to the power of simple lines with a broadcast channel. We show that all three models are of equivalent strength when the security is required to be perfect. If a small probability of failure is allowed, then multicast lines are strictly more powerful than simple lines alone, but equivalent to simple lines with broadcast.

More specifically, we consider two different measures of security: *perfect* (i.e., zero probability that the protocol fails to be secure); and *probabilistic* (i.e., an arbitrarily small probability that the protocol fails to be secure).

We begin this paper by fully exploring the capabilities of multicast lines. Our results for multicast lines are summarized in Fig. 1. Notice that $(t+1)$-connectivity is sufficient to tolerate t arbitrarily malicious faults—closing the connectivity gap between tolerating a passive adversary and an active one that exists for simple lines—if we are willing to tolerate a small probability of error. All of our probabilistically secure protocols have the desirable property that if no faults actually occur, then they will actually provide perfect security.

In Sec. 3, we first consider reliability alone, giving protocols that will be used as building blocks when we consider reliability with privacy. We demon-

reliability	privacy		
	none	probabilistic	perfect
probabilistic	$n > t$ (§3)	$n > t$ (§4.1)	$n > \lceil 3t/2 \rceil$ (§4.2) $n > t$ (§4.3)
perfect	$n > 2t$ (§3)	$n > 2t$ (§4.4)	$n > 2t$ (§4.4)

Fig. 1. Necessary and sufficient connectivity for secure comm over multicast lines

strate a protocol over any $n > t$ multicast lines for transmitting a message with probabilistic security. That is, there remain arbitrarily small probabilities δ and ϵ that the protocol fails to be reliable or private, respectively. The protocol is efficient, in the sense that the round complexity and bit complexity are (low-degree) polynomials in the size of the network, $\log \frac{1}{\delta}$ and $\log \frac{1}{\epsilon}$ (Theorem 5). The main building block for this protocol is an efficient subprotocol for message transmission over $n > t$ multicast lines with probabilistic reliability but with no privacy (Theorem 2). This protocol uses novel authentication techniques for guaranteeing that the correct message "outscores" the wrong ones. We also show (Theorem 3) that perfect reliability over multicast lines cannot be achieved if $n \leq 2t$, providing matching upper and lower bounds.

We then turn to the case of perfect privacy. We modify the probabilistically private protocol to efficiently achieve *perfect* privacy and probabilistic reliability when $n > \lceil 3t/2 \rceil$ (Corollary 8). Using quite different techniques, we can achieve message transmission with perfect privacy and probabilistic reliability over any $n > t$ multicast lines (Theorem 9). However, while the round complexity of the latter protocol is low, its bit complexity is exponential in n. It is open whether an efficient, perfectly private, probabilistically reliable protocol exists for $t < n \leq \lceil 3t/2 \rceil$.

Next, we consider the simple lines model. As shown by [DDWY93], $2t + 1$ simple lines are required for message transmission with perfect security. We show that this remains true if there is also a broadcast channel between sender and receiver (Theorem 14) or if only probablistic security is required (Theorem 11). However, if there is a broadcast channel *and* only probabilistic security is required, then $t+1$ simple lines suffice (Corollary 12). These results, together with comparisons of the multicast model, are summarized in Fig. 2. We remark that it is not immediately obvious that the lower bound techniques for simple lines do not generalize to multicast lines, which makes our $t + 1$ sufficiency results all the more surprising.

Our results can also be used to strengthen the secure multiparty computation result of Rabin and Ben-Or [RB89]. In their setting, $n \geq 2t + 1$ parties are connected by a complete graph of private authenticated single-receiver channels, and also have broadcast. We show that the channel connectivity can be reduced to $t + 1$ in this case (Corollary 13).

	probabilistic security	perfect security
simple lines only	$n > 2t$ (Thm. 11)	$n > 2t$ [DDWY93]
with b/c channel	$n > t$ (Cor. 12)	$n > 2t$ (Thm. 14)
multicast lines	$n > t$ (Thm. 5)	$n > 2t$ (Cor. 10)

Fig. 2. Necessary and sufficient connectivity: comparison of simple and multicast lines

Physical Realizations of Multicast Lines: Suppose that all processors are located on a flat physical plane, and equipped with equally powerful radio transmitter-receivers. Suppose that distances and radio strengths can be adjusted so that all one's immediate neighbors are in radio range (for both receiving and transmitting), while all other processors are out of radio range (for both receiving and transmitting). Suppose that the adversary can change the behavior of processors, but cannot tamper with the radios (e.g., cannot change their strengths or move their locations). In this setting, some number of disjoint multicast lines are realizable, e.g., $n = 2$ disjoint multicast lines between all pairs of processors equidistant around a circle, and $n = 3$ between most pairs of processors on the gridpoints of a hexagonal lattice. To get many disjoint multicast lines from radio broadcast seems to require additional physical assumptions, such as radios tuned to specific frequencies for transmission and reception (which the adversary cannot change), physical barriers to block transmission and reception for certain processors (e.g., rough terrain), or a third dimension for placing transmitter-receivers (e.g., in deep space).

Another way to achieve multicast lines without using radio broadcast is to use overlapping token rings or Ethernet buses: give an active tap to one processor for putting messages onto the ring, and give a passive tap to its immediate neighbors for listening only. This works under the assumption that the adversary can influence the behavior of the faulty processors, but cannot affect the behavior of the physical communication links. Another approach, effective against a polynomially-bounded adversary, is to use broadcast encrypted messages using shared cryptographic keys. Yet another is to rely on a reliable multicast primitive [Ch85] supported by some modern distributed operating systems.

2 The Model

Throughout the paper, n denotes the number of multicast lines and t denotes the number of faults under the control of the adversary. We write $|S|$ to denote the number of elements in the set S. We write $x \in_{\Pr} S$ to indicate that x is chosen with respect to the probability distribution Pr on S, and $x \in_R S$ to indicate a choice with respect to the uniform distribution on S. Our protocols make use of information-theoretically secure authentication over a finite field. For simplicity, we use the same authentication code throughout this paper. Let $\mathbf{F}$ be a finite field, and let $a, b, M \in \mathbf{F}$. We define $\mathrm{auth}(M, a, b) = aM + b$. This function has been previously used for similar purposes (cf. [RB89,Rab94]).

Communication Model: Party A (the message transmitter) and party B (the message recipient) are connected by n *lines*. The jth line is a sequence of $m+2$ nodes $X_{0,j}, X_{1,j}, \ldots, X_{m,j}, X_{m+1,j}$, where $X_{0,j} = A$ and $X_{m+1,j} = B$, $m \geq 1$. We may use the ordered pair (i,j) to denote the node $X_{i,j}$, and V to denote the set of all nodes $\{(i,j) : 0 \leq i \leq m+1, 1 \leq j \leq n\}$. Let $G = (V, E)$ be the undirected graph with edges $E = \{(X_{i,j}, X_{i+1,j}) : 0 \leq i \leq m, 1 \leq i \leq n\}$, i.e., neighbors on a line are neighbors in G. We may use the term *internal node* to denote $V - \{A, B\}$. (Note that it is clear how to modify all our protocols and lower bound proofs to the case where lines are of different lengths.)

We consider *multicast* as our only communication primitive. A message that is multicast by any node is received by all its neighbors (i.e., both neighbors of an internal node, or all n neighbors of A or B). Furthermore, a multicast value is received with privacy (i.e., non-neighbors learn nothing about what was sent) and authentication (i.e., neighbors are guaranteed to receive the value that was multicast and to know which neighbor multicast it).

In a *message transmission protocol*, the sender A starts with a message M^A drawn from a message space $\mathcal{M}$ with respect to a probability distribution Pr. At the end of the protocol, the receiver B outputs a message $M^B \in \mathcal{M}$. We consider a synchronous system in which messages are sent via multicast in *rounds*. During each round of the protocol, each node receives any messages that were multicast by its neighbors at the end of the previous round, flips coins and performs local computations, and then possibly multicasts a message. For all of the protocols in this paper, $\mathcal{M}$ must be representable as a subset of a finite field $\mathbf{F}$.

Adversary Model: We consider *active*, or *Byzantine*, attacks, in which t internal nodes are under the control of an adversary of unlimited computational power. The adversary is assumed to know the complete protocol specification, message space $\mathcal{M}$, size of the network, and any inputs other than M^A held by any party (i.e., all relevant information except M^A and the coin flips used by V during the execution). At the start of the protocol, the adversary chooses the message distribution Pr and the t faulty nodes. It is a simplifying assumption that all faults are chosen before the start of the protocol, but the results in this paper are not affected if the adversary is given the additional power to choose faults during the execution of the protocol. The adversary can view all the behavior of the faulty nodes (coin flips, computations, messages received) as well as control the messages that they multicast. The adversary cannot violate the multicast constraint, i.e., whatever is received by one neighbor of a faulty node is received by both neighbors.

For any execution of the protocol, let adv be the adversary's view of the entire protocol, i.e., the behavior of the faulty nodes in every round, the initial state of the adversary, and the coin flips of the adversary in every round. We write $\mathrm{adv}(M, r)$ to denote the adversary's view when $M^A = M$ and when the sequence of coin flips used by the adversary is r. Note that adv and $\mathrm{adv}(M, r)$ are random variables, e.g., $\mathrm{adv}(M, r)$ depends on the coin flips of the honest parties.

Reliability: A message transmission protocol is δ-*reliable* if, with probability at least $1-\delta$, B terminates with $M^B = M^A$. The probability is over the choice of M^A and the coin flips of V and the adversary.

Privacy: A message transmission protocol is ϵ-*private* if, for every two messages $M_0, M_1 \in \mathcal{M}$ and every r, $\sum_c |\Pr[\text{adv}(M_0, r) = c] - \Pr[\text{adv}(M_1, r) = c]| \leq 2\epsilon$. The probabilities are taken over the coin flips of the honest parties, and the sum is over all possible values of the adversary's view.

Security: A message transmission protocol is (ϵ, δ)-*secure* if it is δ-reliable and ϵ-private.

Efficiency: An (ϵ, δ)-secure message transmission protocol is *efficient* if its round complexity and bit complexity are polynomial in the size of the network, $\log \frac{1}{\epsilon}$ (if $\epsilon > 0$), and $\log \frac{1}{\delta}$ (if $\delta > 0$).

Note that if $t \geq n$, then it is possible to achieve neither reliable nor secure message transmission, since an adversary can place one fault on each line and either block or monitor all communication between A and B. We therefore assume $t < n$ throughout the remainder of the paper.

3 Reliable Communication Over Multicast Lines

In this section, we address the question of reliable communication, with no requirement of privacy. We first consider probabilistic reliability and show in Theorem 2 that it is achievable whenever $n > t$. We also show in Theorem 3 that perfect reliability is possible only when $n > 2t$.

In this section, we show how to efficiently achieve δ-reliable communication for $\delta > 0$ when $n > t$. To achieve reliable communication, we use two subprotocols. In the Basic Propagation Protocol, A tries to propagate a value s^A from a set S to B. To do this, the multicast lines are used essentially as simple lines. First, A sends s^A to its neighbors. In turn, each (non-faulty) node receives and propagates s^A "down" the simple line towards B.

Basic Propagation Protocol
- In round 1, A multicasts s^A.
- In round ρ for $2 \leq \rho \leq m$, each $X_{\rho-1,j}$ $(1 \leq j \leq n)$ expects to receive a single element of S from $X_{\rho-2,j}$. Let $u_{\rho-1,j}$ be this value if a value was in fact received, or a publicly known default element otherwise. At the end of round ρ, $X_{\rho-1,j}$ multicasts $u_{\rho-1,j}$.
- In round $m+2$, B receives a single element of S from each $X_{m,j}$, or substitutes the default element. Let s_j^B be the value received or substituted on line j.

In the Full Distribution Protocol, each internal node $X_{i,j}$ tries to transmit an element $s_{i,j} \in S$ to A and B. As in the Basic Propagation Protocol, the lines are used essentially as simple lines. In order to keep track of which messages should be propagated, the "intended" recipient or recipients of a message are included.

Full Distribution Protocol

– In round 1, each $X_{i,j}$ multicasts $(s_{i,j}, \{X_{i-1,j}, X_{i+1,j}\})$.
– In round ρ for $2 \leq \rho \leq m+1$:
  - For $1 \leq j \leq n$ and $\rho \leq i \leq m$, $X_{i,j}$ expects to be the intended recipient of an element from $X_{i-1,j}$ (initiated by $X_{i-\rho+1,j}$). Let $u_{i,j}$ be the received value or a default value if none is received.
  - For $1 \leq j \leq n$ and $1 \leq i \leq m-\rho+1$, $X_{i,j}$ expects to be the intended recipient of an element from $X_{i+1,j}$ (initiated by $X_{i+\rho-1,j}$). Let $v_{i,j}$ be the received or default value.
  - For $1 \leq j \leq n$, A expects to be the intended recipient on the jth line of a single element (initiated by $X_{\rho-1,j}$). Let $s^A_{\rho-1,j}$ be the received or default value.
  - For $1 \leq j \leq n$, B expects to be the intended recipient on the jth line of a single element from $X_{m+\rho,j}$. Let $s^B_{m+\rho,j}$ be the received or default value.
  - $X_{i,j}$ multicasts $(u_{i,j}, X_{i+1,j})$ if $\rho \leq i \leq m$, and $(v_{i,j}, X_{i-1,j})$ if $1 \leq i \leq m-\rho+1$.

Fact 1. *If there are no faults on the jth line, then $s^A_{i,j} = s^B_{i,j} = s_{i,j}$ for all $1 \leq i \leq m$. Further, if $X_{i,j}$ is the only fault on the jth line, then $s^A_{i,j} = s^B_{i,j}$.*

To achieve reliable message transmission, each internal node chooses a random authentication key. A's message M^A is authenticated with respect to each of these mn random authentication keys. The adversary can only reliably forge an authentication if it has seen the key, i.e., for keys initiated on a line with at least one fault. By contrast, A and B agree on at least one authentication key from each fault-free and single-fault line. If all received messages are ranked by B according to the number of lines from which corroborating authentication keys originated, then the real message will almost always get the highest rank.

Reliable Transmission Protocol

– In rounds 1 through $m+2$, the nodes of V execute an instance of the Full Distribution Protocol. The element that $X_{i,j}$ initiates is $(a_{i,j}, b_{i,j}) \in_R \mathbf{F}^2$. Let $(a^A_{i,j}, b^A_{i,j})$ and $(a^B_{i,j}, b^B_{i,j})$ be the values that A and B receive or substitute as the element initiated by $X_{i,j}$.
– In rounds $m+3$ through $2m+4$, the nodes of V execute an instance of the Basic Propagation Protocol from A to B. The element that A initiates is $\{(i, j, M^A, \mathrm{auth}(M^A, a^A_{i,j}, b^A_{i,j})) : 1 \leq i \leq m, 1 \leq j \leq n\}$. In round $2m+4$, B receives or substitutes $\{(i, j, M^B_{i,j,k}, u^B_{i,j,k}) : 1 \leq i \leq m, 1 \leq j \leq n\}$ on the kth line, $1 \leq k \leq n$.
– Let $r_k(M) = |\{j : \exists i. M = M^B_{i,j,k}$ and $u^B_{i,j,k} = \mathrm{auth}(M^B_{i,j,k}, a^B_{i,j}, b^B_{i,j})\}|$. B outputs $M^B \in \mathbf{F}$ that maximizes $\max_k r_k(M^B)$.

Note that in the last round, B need only check messages that were actually received on each line k as $M^B_{i,j,k}$ for some i, j, and not all the elements of $\mathbf{F}$.

The Reliable Transmission protocol provides δ-reliable message transmission provided that the field $\mathbf{F} \supseteq \mathcal{M}$ used by auth() satisfies $|\mathbf{F}| > \frac{mn^2}{\delta}$. Since reliable communication is not possible when $t > n$, this protocol provides matching upper and lower bounds for probabilistic reliability without privacy.

Theorem 2. *If $\delta > 0$ and $n > t$, the Reliable Transmission Protocol is an efficient δ-reliable message transmission protocol.*

Proof (sketch): Let w_0 denote the number of lines with no faults, w_1 the number with exactly one fault, and w_+ the number with two or more faults. Then since $n > t$, it follows that $w_0 > w_+$. In addition, by Fact 1, there exists k such that $r_k(M^A) \geq w_0 + w_1$. Further, it follows from the use of auth() that the probability that there exist k and $M \neq M^A$ such that $r_k(M) > w_1 + w_+$ is less than $mn^2/|\mathbf{F}|$. Taking $\mathbf{F}$ such that $|\mathbf{F}| > mn^2/\delta$, it follows that $\Pr[M^B = M^A] > 1 - \delta$. □

Theorem 3 shows that perfect reliability is unachievable over n multicast lines when $n \leq 2t$. The proof follows Dolev et al. [DDWY93].

Theorem 3. *0-reliable message transmission over n multicast lines is impossible when $n \leq 2t$.*

Proof: Consider a graph of $n = 2t$ multicast lines, each of length $m \geq 1$, and suppose that Π is a message transmission protocol. The adversary behaves as follows. The adversary flips a coin to decide whether to disrupt $W_0 = \{X_{1,1}, \ldots, X_{1,t}\}$ or $W_1 = \{X_{1,t+1}, \ldots, X_{1,2t}\}$. Let W_b denote the faulty subset, and let W_{1-b} denote the honest subset.

Let s_ρ^{ij} be the message multicast by processor $X_{i,j}$ in round ρ of the execution. Let s_ρ^A (respectively s_ρ^B) be the message multicast by A (respectively B) in round ρ of the execution. Let $\widehat{s}_\rho^A$ be the message, chosen by the adversary, that A supposedly multicast in round ρ of the simulation. In each round ρ, the adversary causes each $X_{1,j}$ in W_b to follow the protocol Π as if the messages that it received from A were $\widehat{s}_1^A, \ldots, \widehat{s}_{\rho-1}^A$. That is, the message s_ρ^{1j} that the faulty $X_{1,j}$ multicasts in round ρ is a function of the simulated messages $\widehat{s}_1^A, \ldots, \widehat{s}_{\rho-1}^A$, the real messages $s_1^{2j}, \ldots, s_{\rho-1}^{2j}$ from X_{2j}, and local coin flips for $X_{1,j}$ chosen at random by the adversary.

With nonzero probability, all of the adversary's choices for $\widehat{s}_1^A, \ldots, \widehat{s}_\rho^A$ are consistent with a possible behavior of A executing Π for some other message, so B cannot halt at the end of round ρ and output M^B with certainty. □

4 Secure Communication Over Multicast Lines

In this section, we consider reliable *and* private communication. By Theorem 3, we can not hope to achieve perfect reliability unless $n > 2t$. Hence, we first consider the case of probabilistic privacy with probabilistic reliability. We show in Sec. 4.1 that probabilistic security is achievable whenever $n > t$. In Sec. 4.2, we show that it is possible to efficiently achieve perfect privacy with probabilistic reliability when $n > \lceil 3t/2 \rceil$. We do not know whether it is possible to efficiently achieve perfect privacy when $t < n \leq \lceil 3t/2 \rceil$, but we are able to give an inefficient solution in Sec. 4.3. In Sec. 4.4, we point out that the protocol of [DDWY93], combined with our protocols, can be modified to work for perfect privacy with perfect reliability over multicast lines if $n > 2t$.

4.1 Probabilistic Security

In the Private Propagation Protocol, A tries to propagate a different $s_j^A \in S$ to B on each line j, $1 \leq j \leq n$. This protocol demonstrates why it does not matter whether the multicast property is extended to sender and receiver in our model.

> *Private Propagation Protocol*
> - In round 1, each $X_{1,j}$ multicasts $r_j \in_R S$.
> - In round 2, A multicasts $(u_1, \ldots, u_n)$, where each $u_j = s_j^A + r_j$.
> - In rounds 3 through $m+4$ each $X_{1,j}$ now proceeds as in the Basic Propagation Protocol with the value $s_j = u_j - r_j$. Let s_j^B be the element ultimately received by B on the jth line.

Fact 4. *If there are no faults on the jth line, then $s_j^B = s_j^A$ and* $\Pr[s_j^A = s|\mathrm{adv}] = \Pr[s_j^A = s]$.

Using the Private Propagation Protocol, we can achieve private message transmission. Intuitively, the protocol works as follows. A privately propagates a different random one-time pad on each line to B. Using the Reliable Transmission Protocol from the preceding section and a randomized authentication procedure, A and B determine which pads have been received identically at both ends. A then encrypts the message using the sum of the pads that pass the test, and transmits this encryption reliably (and non-privately) to B. A similar protocol appears in [BF97]. Formally, we have the following:

> *Private Transmission Protocol*
> - In rounds 1 through $m+4$, the nodes of V execute an instance of the Private Propagation Protocol. A propagates to B the values $c_j^A, d_j^A \in_R \mathbf{F}^2$ on each line j. Let c_j^B, d_j^B be the values received by B on the j line.
> - For each j, B chooses $r_j^B \in_R \mathbf{F}$, and computes $s_j^B = \mathrm{auth}(r_j^B, c_j^B, d_j^B)$. In rounds $m+5$ through $3m+9$, the nodes of V execute an instance of the Reliable Transmission Protocol. B $(\min(\epsilon, \frac{\delta}{3}))$-reliably transmits to A the values r_j^B, s_j^B. Let r_j^A, s_j^A for $1 \leq j \leq n$ be the values received by A as the output of the Reliable Transmission Protocol.
> - A computes $W^A = \{j : s_j^A = \mathrm{auth}(r_j^A, c_j^A, d_j^A)\}$ and $z^A = M^A + \sum_{j \in W^A} c_j^A$. In rounds $3m+10$ through $5m+13$, the nodes of V execute another instance of the Reliable Transmission Protocol. A $\frac{\delta}{3}$-reliably transmits to B the values W^A and z^A. Let W^B, z^B be the values received by B as the output of the Reliable Transmission Protocol.
> - B computes $M^B = z^B - \sum_{j \in W^B} c_j^B$.

By choosing $\mathbf{F}$ such that $|\mathbf{F}| \geq \frac{3n}{\delta}$, it follows that the Private Transmission Protocol is efficient, private, and reliable.

Theorem 5. *If $\epsilon > 0$, $\delta > 0$, and $n > t$, the Private Transmission Protocol is an efficient (ϵ, δ)-secure message transmission protocol.*

Proof (sketch): To see that the Private Transmission Protocol is ϵ-private, let j^* be a non-faulty line. Then $c^A_{j^*} = c^B_{j^*}$ and $d^A_{j^*} = d^B_{j^*}$. Let RT denote the event that the reliable transmission from B to A succeeds, and suppose that RT occurs. Then $s^A_{j^*} = s^B_{j^*} = c^B_{j^*} r^B_{j^*} + d^B_{j^*} = c^A_{j^*} r^A_{j^*} + d^A_{j^*}$, and so $j^* \in W^A$. Since $z^A = M^A + c^A_{j^*} + \sum_{j \in W^A, j \neq j^*} c^A_j$, we have that every M^A is equally likely given r^A_j, s^A_j, z^A. Since this is the only relevant information about M^A in adv, we have that $\Pr[\mathrm{adv}(M_0, r) = c | \mathrm{RT}] = \Pr[\mathrm{adv}(M_1, r) = c | \mathrm{RT}]$ for every pair of messages M_0 and M_1, adversary coin flips r, and possible view c. Let C_i be the set of adversary views where $M^A = M_i$ and RT succeeded; let $\bar{C}_i$ be the set of adversary views where $M^A = M_i$ and RT failed. Then $\Pr[\mathrm{RT} | M^A = M, r] \geq 1 - \epsilon$ for all M and all adversary coin flips r. Thus, $\sum_{c \in C_i} | \Pr[\mathrm{adv}(M_0, r) = c] - \Pr[\mathrm{adv}(M_1, r) = c]| = 0$ and $\sum_{c \in \bar{C}_i} | \Pr[\mathrm{adv}(M_0, r) = c] - \Pr[\mathrm{adv}(M_1, r) = c]| \leq \epsilon$. It follows that $\sum_c | \Pr[\mathrm{adv}(M_0, r) = c] - \Pr[\mathrm{adv}(M_1, r) = c]| \leq 2\epsilon$.

Next, suppose both reliable transmissions succeed. Then $r^A_j = r^B_j$ and $s^A_j = s^B_j$ for all $1 \leq j \leq n$, and $W^A = W^B$ and $z^A = z^B$. Therefore, if $j \in W^A$, then $c^A_j r^A_j + d^A_j = s^A_j = s^B_j = c^B_j r^B_j + d^B_j = c^B_j r^A_j + d^B_j$ which implies that $r^A_j = (d^B_j - d^A_j)(c^A_j - c^B_j)^{-1}$. Since c^A_j, d^A_j, c^B_j, and d^B_j are fixed before the random choice of r^A_j, it follows that for any fixed $j \in W^A$, $\Pr[c^A_j \neq c^B_j] \leq 1/|\mathbf{F}|$. If both reliable transmissions succeed and $M^B \neq M^A$, then $c^A_j \neq c^B_j$ for at least one $j \in W^A$. By the above, this occurs with probability at most $|W^A|/|\mathbf{F}| < n/|\mathbf{F}|$. Take $\mathbf{F} \supseteq \mathcal{M}$ such that $|\mathbf{F}| \geq \frac{3n}{\delta}$. Then by Theorem 2, the probability that either reliable message transmission fails is no more than $\frac{2\delta}{3}$. Therefore, $\Pr[M^A \neq M^B] \leq \frac{2\delta}{3} + \frac{n}{|\mathbf{F}|} \leq \delta$. □

Since secure communication is not possible when $t > n$, this protocol provides matching upper and lower bounds for probabilistic privacy with probabilistic reliability.

4.2 Perfect Privacy when $n > \lceil 3t/2 \rceil$

Note that the requirement that $\epsilon > 0$ is necessary since the second step of the protocol requires a $(\min(\epsilon, \frac{\delta}{3}))$-reliable transmission. In fact, there is an adversary attack against the protocol that succeeds in compromising privacy with nonzero (at most ϵ) probability. First, the adversary listens on t lines in the private propagation phase. The adversary then partially disrupts the first reliable transmission from B to A, affecting on each of the t faulty lines all of the values associated with the fault-free lines. If the adversary successfully guesses the appropriate unseen authentication in the Reliable Transmission Protocol, the disruption succeeds, and no authentication check by A passes for any fault-free line. In this case, W^A contains only faulty lines, allowing the adversary to determine M^A from z^A.

It is possible to foil this attack if the first reliable transmission from B to A is done using $\frac{\delta}{3}$-reliable message transmission such that A can detect when the correct message is not received. Then A could send nothing when this reliable transmission fails. In fact, it is easy to show that the adversary never learns *any* information about the message.

Fortunately, the Reliable Transmission Protocol of Sec. 3 can easily be modified to provide this property when $n > \lceil 3t/2 \rceil$:

Definition 6. *A message transmission protocol is* perfectly detecting *if B either terminates with $M^B = M^A$ or terminates and outputs nothing.*

Corollary 7 (to Theorem 2). *If $\delta > 0$ and $n > \lceil 3t/2 \rceil$, then there exists an efficient perfectly-detecting δ-reliable message transmission protocol.*

Proof: Add another condition to the output rule for B in the final step of the Reliable Transmission Protocol. In order for B to output M^B, we will also require that $\max_k r_k(M) \leq t$ for every other $M \neq M^B$. If there are two or more messages with rank greater than t, then B will terminate and output nothing. If $n > \lceil 3t/2 \rceil$, then B will always either output the correct message or will output nothing, and so the modified protocol is perfectly detecting. $\square$

Corollary 8. *If $\delta > 0$ and $n > \lceil 3t/2 \rceil$, then there exists an efficient $(0, \delta)$-secure message transmission protocol.*

4.3 Perfect Privacy when $t < n < \lceil 3t/2 \rceil$

In Sec. 4.2, we showed how to efficiently achieve perfect privacy and probabilistic reliability when $n > \lceil 3t/2 \rceil$. In this section, we continue our investigation of perfect privacy, and show that perfect privacy and probabilistic reliability can be achieved at minimum connectivity of $n > t$, although the bit complexity is exponential in the number of lines. We do not know whether an efficient solution exists when $t < n \leq \lceil 3t/2 \rceil$.

Intuitively, our protocol proceeds as follows. The receiver attempts to transmit to the sender many random, uniquely labeled, one-time pads. The sender is able to find one pad that was transmitted with perfect privacy and probabilistic reliability. The sender then transmits to the receiver—with probabilistic reliability *and without privacy*—the encryption of the message using the one-time pad, together with the label of the pad. The receiver can look up the one-time pad from the label, and decrypt the message.

Formally, define a *probe set* S to be a subset of nodes such that no two nodes are in the same line: If $(i, j) \in S$ and $i' \neq i$ then $(i', j) \notin S$. Let $\mathcal{L}$ denote the set of all probe sets. Let $\psi : \mathcal{L} \rightarrow \mathbf{F}$ be an injective mapping from probe sets to elements of $\mathbf{F}$. Given a function $f(x) = (y_1, y_2, y_3)$, we write $f_i(x)$ to denote y_i. We define a *double masking* procedure for authentication with secrecy:

$$\text{DoubleMask}(M, a, b, c) = (aM + b, M + c)$$

Without knowledge of the secret key, no information about the encrypted value can be inferred, and any tampering with the ciphertext is almost always detected. We define the corresponding *unmask* procedure:

$$\text{Unmask}((u, v), a, b, c) = \begin{cases} v - c & \text{if } a(v - c) = (u - b) \\ \text{undefined otherwise} \end{cases}$$

Then the protocol is as follows:

Perfectly Private Transmission Protocol

- In rounds 1 through $m + 1$, the nodes of V execute an instance of the Full Distribution Protocol. The element that $X_{i,j}$ initiates is $f_{i,j} : \mathcal{L} \to \mathbf{F}^3$, chosen uniformly at random from the set of all complete functions from $\mathcal{L}$ to $\mathbf{F}^3$. Let $f^A_{i,j}$ and $f^B_{i,j}$ be the elements received by A and B, respectively, corresponding to the element initiated by $X_{i,j}$. Let $\alpha(S) = \sum_{(i,j)\in S} f^A_{i,j}(S)$, and $\beta(S) = \sum_{(i,j)\in S} f^B_{i,j}(S)$, for every $S \in \mathcal{L}$.
- B computes $g^B : \mathcal{L} \to \mathbf{F}^2$, where $g^B(S) = \text{DoubleMask}(r^B_S, \beta_1(S), \beta_2(S), \beta_3(S))$, $r^B_S \in_R \mathbf{F}$, for every $S \in \mathcal{L}$. In rounds $m+2$ through $2m+3$, B propagates g^B to A using the Basic Propagation Protocol. Let g^A_j be the element that A receives on the jth line in round $2m + 3$.
- A computes $z^A = (\psi(S'), M^A + r^A_{S',j})$, where $r^A_{S',j} = \text{Unmask}(g^A_j(S'), \alpha_1(S'), \alpha_2(S'), \alpha_3(S'))$, and no larger probe set leads to a successful unmasking for any j. In rounds $2m + 4$ through $4m + 7$, A sends z^A to B using the (probabilistically) Reliable Transmission Protocol. Let $z^B = (x^B, y^B)$ be the element that B accepts as the outcome of the Reliable Transmission Protocol.
- B outputs $M^B = y^B - r^B_{\psi^{-1}(x^B)}$.

When $\mathbf{F} \supseteq \mathcal{M}$ such that $|\mathbf{F}| > (1/\delta)\,(n(m+1)^n + 1/2mn^2)$, this protocol achieves perfect privacy and probabilistic reliability.

Theorem 9. *Let $\delta > 0$ and $n > t$. Then the Perfectly Private Transmission Protocol is $(0, \delta)$-secure.*

Note, however, that the protocol is not efficient, since the message sent in the first step is the description of a function on the set of probe sets, which is of size $(m + 1)^n$.

4.4 Perfect Security

To complete our treatment of secure communications over multicast lines, we note that it is easy to achieve perfectly secure message transmission over $n >$

$2t$ multicast lines by using the Private Propagation Protocol from Sec. 4.1 to simulate the protocol of [DDWY93] for $n > 2t$ simple lines.

Corollary 10. *If $n > 2t$, then there exists an efficient $(0, 0)$-secure message transmission protocol.*

Note that this protocol can also be used for probabilistic privacy with perfect reliability, so we have now addressed all combinations of reliability and privacy.

5 Secure Communication Without Multicast

In this section, we compare the multicast model to simple lines with and without broadcast. We say that there are n simple lines connecting sender and receiver if they are connected by n disjoint paths of private, authenticated, single-receiver channels. We say that there is broadcast if any party can send an authenticated message that will be received by all parties.

5.1 Simple Lines

Dolev et al. [DDWY93] showed that $2t+1$ simple lines are necessary and sufficient for perfectly secure message transmission. We showed in Sec. 3 that, similarly, $2t + 1$ multicast lines are necessary and sufficient for perfectly secure message transmission. However, as shown in Sec. 4.1, only $t+1$ multicast lines are needed for probabilistically secure message transmission. In contrast, we show in this section that the $2t+1$ bound in the simple lines model holds even for probabilistic security. Thus, multicast lines are strictly more powerful than simple lines alone when a small probability of failure is allowed, but are equivalent to simple lines if no failure is allowed.

Specifically, we show that $2t + 1$ simple lines are required for reliable message transmission even if we allow a substantial probability of failure. It is easy to achieve $1/2$-reliability when $n = 2t$: send M^A on all lines, and have B take a majority vote, where B uses a coin flip to break a t-to-t tie. Theorem 11 shows that it is not possible to do substantially better (proof omitted).

Theorem 11. *If $n \leq 2t$ and $\delta < \frac{1}{2}(1 - \frac{1}{|\mathcal{M}|})$, then δ-reliable message transmission over n simple lines is impossible.*

5.2 Simple Lines with Broadcast

Anything done over simple lines can be simulated over multicast lines using the Private Propagation Protocol (Sec. 4.1), and vice versa. A similar relationship holds between broadcast and the Reliable Transmission Protocol (Sec. 3). This allows translation of the Private Transmission Protocol to the setting of simple lines with broadcast. It is somewhat unintuitive that the translated protocol achieves perfect *privacy* but is still only probabilistically *reliable*, since an adversary can still disrupt the private propagation and cause the receiver to output the wrong message with nonzero probability.

Corollary 12. *$(0,\delta)$-secure communication is possible over $n > t$ simple lines with broadcast.*

Implications for Secure Multiparty Computation: Corollary 12 can be used to strengthen the secure multiparty computation result of Rabin and Ben-Or [RB89]. In their setting, $n > 2t+1$ parties are connected by a complete graph of private, authenticated, single-receiver channels, and also any player can broadcast a message that will be received authentically by all players. The channel connectivity can be reduced to $t+1$, since the $(0,\delta)$-protocol from Corollary 12 can simulate the missing channels. The small probability δ that each simulation fails is not significant, since the protocol of Rabin and Ben-Or already has a negligible probability of failure. Indeed, this error is necessary, since error-free multiparty computation requires $3t+1$ connectivity [BGW88,CCD88,RB89].

Corollary 13. *Secure multiparty computation, with an arbitrarily small probability of error, is efficient over a $(t{+}1)$-connected network of private authenticated channels with broadcast.*

One might hope that the broadcast channel would avoid the $n > 2t$ connectivity requirement for perfect security. Theorem 14 shows that this is not the case (proof omitted). Together with Corollary 12, this shows that multicast lines and simple lines with broadcast are equivalent for secure communication.

Theorem 14. *$(0,0)$-secure message transmission over n simple lines with a broadcast channel is impossible if $n \leq 2t$.*

6 Conclusions

We have considered the problem of secure communication over multicast lines. We have given a complete characterization of when it is possible to give a solution, and an almost complete characterization of when it is possible to give an efficient solution. The question remains open whether there exists an efficient $(0,\delta)$-secure message transmission protocol when $t < n \leq \lceil 3t/2 \rceil$.

In addition, we compared multicast lines to the simple lines alone or with broadcast. We showed that all three models are of equivalent strength when the security is required to be perfect, while multicast lines and simple lines with broadcast are more powerful than simple lines alone when security need only be probabilistic. In particular, our results yield improved protocols for secure multiparty computation in a network of private authenticated channels with broadcast, reducing the necessary connectivity to $t+1$.

In all of the multicast protocols described in this paper, the multicast property is only *needed* to multicast values drawn from a uniform distribution. With simple modifications, the protocols would retain their security properties in a communication setting that had multicast lines for the first round and simple lines thereafter. This suggests that there may be a more fundamental “atom” than multicast for establishing secure communication with low connectivity.

A more general setting is a multicast graph, with a channel from each node to its neighborhood. If a graph has n disjoint paths whose neighborhoods are also disjoint, then our multicast lines protocols can be simulated on the multicast graph. However, if these n disjoint paths do not have disjoint neighborhoods, then an adversary may be able to foil our protocols with $t < n$ faults by using one fault to eavesdrop on two disjoint lines. An obvious direction of further research would be to fully characterize secure communication in this more general setting.

References

[ABLP89] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg, "On the complexity of radio communication," *ACM STOC*, 1989, 274–285.

[Bea89] D. Beaver, "Multiparty protocols tolerating half faulty processors," *Proc. Crypto '89*, 560–572.

[BF97] A. Beimel and M. Franklin, "Reliable communication over partially authenticated networks," *Proc. Workshop on Distributed Algorithms (WDAG)*, Springer-Verlag LNCS 1320, 1997, 245–259.

[BCG93] M. Ben-Or, R. Canetti, and O. Goldreich, "Asynchronous secure computation," *ACM STOC*, 1993, 52–61.

[BGW88] M. Ben-Or, S. Goldwasser, and A. Wigderson, "Completeness theorems for non-cryptographic fault-tolerant distributed computing," *ACM STOC*, 1988, 1–10.

[BT85] G. Bracha and S. Toueg, "Asynchronous consensus and broadcast protocols", *JACM* 32(4): 824–840 (1985).

[BDK97] M. Burmester, Y. Desmedt, and G. Kabatianski, "Trust and Security: A New Look at the Byzantine Generals Problem", *Network Threats*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, (to appear).

[CCD88] D. Chaum, C. Crepeau, and I. Damgard, "Multiparty unconditional secure protocols," *ACM STOC*, 1988, 11–19.

[Ch85] D. Cheriton and W. Zwaenepoel, "Distributed process group in the V kernel" *ACM Trans. on Computer Systems* 3, (1985), 77–107.

[Dol82] D. Dolev, "The Byzantine Generals strike again," *J. Algorithms* 3:14–30, 1982.

[DDWY93] D. Dolev, C. Dwork, O. Waarts, and M. Yung, "Perfectly secure message transmission," *JACM* 40(1):17–47, 1993.

[FY95] M. Franklin and M. Yung, "Secure hypergraphs: privacy from partial broadcast," *ACM STOC*, 1995, 36–44.

[GGL91] O. Goldreich, S. Goldwasser, and N. Linial, "Fault-tolerant computation in the full information model," *IEEE FOCS*, 1991, 447–457.

[PSL80] M. Pease, R. Shostak, and L. Lamport, "Reaching Agreement in the Presence of Faults", *JACM* 27(2): 228–234, 1980.

[PG89] F. M. Pittelli and H. Garcia-Molina, "Reliable Scheduling in a TMR Database System", *ACM Transactions on Computer Systems* 7(1): 25–60, 1989.

[Rab94] T. Rabin, "Robust sharing of secrets when the dealer is honest or faulty," *JACM* 41(6):1089–1109, 1994.

[RB89] T. Rabin and M. Ben-Or, "Verifiable secret sharing and multiparty protocols with honest majority," *ACM STOC*, 1989, 73–85.