Efficient Reliable Communication over Partially Authenticated Networks

Amos Beimel
Computer Science Department
Ben Gurion University, Beer Sheva 84105, Israel.
beimel@cs.bgu.ac.il

Lior Malka
Computer Science Department
Ben Gurion University, Beer Sheva 84105, Israel.
liorma@cs.bgu.ac.il

ABSTRACT

Reliable communication between parties in a network is a basic requirement for executing any protocol. Dolev [4] and Dolev et al. [5] showed that reliable communication is possible if and only if the communication network is sufficiently connected. Beimel and Franklin [1] showed that the connectivity requirement can be relaxed if some pairs of parties share authentication keys. That is, costly communication links can be replaced by authentication keys.

In this work, we continue this line of research. We consider the scenario where there is a specific sender and a specific receiver. In this case, the protocol of [1] has $n^{O(n)}$ rounds even if there is a single Byzantine processor. We present a more efficient protocol with round complexity of $(n/t)^{O(t)}$, where n is the number of processors in the network and t is an upper bound on the number of Byzantine processors in the network. Specifically, our protocol is polynomial when the number of Byzantine processors is $O(1)$, and for every t its round complexity is bounded by $2^{O(n)}$. The same improvements hold for reliable and private communication. The improved protocol is obtained by analyzing the properties of a "communication and authentication graph" that characterizes reliable communication.

1. INTRODUCTION

Suppose that some processors are connected by a network of reliable channels. All of the processors cooperate to execute some protocol, but some of them are maliciously faulty. Dolev [4] and Dolev et al. [5] proved that if there are t faulty processors, then every pair of processors can communicate reliably if and only if the network is $(2t + 1)$-connected. Beimel and Franklin [1] showed that the connectivity requirement can be relaxed if some pairs of parties share authentication keys. That is, instead of costly communication channels, we can give some pairs of processors (other than the pairs connected by channels) authentication keys, i.e., the means to identify messages from the other.

In this paper we consider the problem of "single-pair" reliable communication. In this problem there is a specific sender who wants to send a message to a specific receiver, such that any coalition of at most t faulty processors cannot prevent this transmission.

PODC 2003, Boston MA

The network of channels defines a natural "communication graph," with an edge between two vertices for every channel between two processors. The pairs of parties sharing authentication keys define a natural "authentication graph," with an edge between two vertices for every shared key. Beimel and Franklin [1] characterize when reliable communication is possible using these two graphs; their characterization depends on recursively defined graphs which include all the edges of the communication graph and some of the edges of the authentication graph. However, the reliable protocol presented by Beimel and Franklin [1] is inefficient; it requires $n^{O(n)}$ rounds, where n is the number of processors in the network. In this paper we present a more efficient protocol obtained by exploiting the properties of the graphs that characterize reliable communication.

Historical Notes. The connectivity requirements for several distributed tasks in several models has been studied in many papers; for example Byzantine agreement [4, 8], approximate Byzantine agreement [6, 16], reliable message transmission [4, 5], and reliable and private message transmission [13, 5, 14]. Simple impossibility results and references can be found in [8, 12]. We mention that in Byzantine agreement all honest parties should agree on the same message while in reliable communication only the transmitter and the receiver agree on the message. Beimel and Franklin [1] considered the connectivity requirements in partially authenticated networks. In addition to the "single-pair" version of the problem, they characterize when reliable transmission is possible in the "all-pairs" version. In this version any transmitter should be able to reliably transmit a message to any receiver, such that any coalition of at most t faulty processors cannot prevent this transmission. Goldreich, Goldwasser, and Linial [11], Franklin and Yung [10], Franklin and Wright [9], and Wang and Desmedt [2] have studied secure communication and secure computation in multi-recipient (multi-cast) models. Wang and Desmedt [3] studied secure computation in directed networks.

Our Results. Our main result is a more efficient protocol for "single-pair" reliable communication. The round complexity of our protocol is $(n/t)^{O(t)}$, where n is the number of processors in the network and t is an upper bound on the number of Byzantine processors in the network. Specifically, our protocol is polynomial when the number of Byzantine processors is $O(1)$, and for every t its round complexity is bounded by $2^{O(n)}$. The improved protocol is obtained by analyzing the properties of the graphs that characterize reliable communication. We exploit these properties to show that there is an implementation of the protocol of [1] with better round complexity.

Our results have implications to reliable and *private* communication, also known as secure communication, that is, communication in which any coalition of at most t faulty processors cannot learn any information on the message that is being sent and cannot prevent it. In [1] it is shown that reliable and private communication from a to b is possible if and only if reliable communication from a to b and from b to a are possible. Thus, our results translate into more efficient reliable and private communication.

We also give a simple characterization for reliable communication against one Byzantine processor. We show that in this case a simple necessary condition, namely that the communication graph is 2-connected between a and b and the union of the communication and authentication graphs is 3-connected between a and b is "basically" sufficient. This characterization implies that reliable communication is symmetric for $t = 1$. However, we show that the natural generalization of this condition to $t \geq 2$ is not sufficient. Finally, we show that reliable communication is not symmetric for $t \geq 2$. That is, there is a communication graph and an authentication graph for which reliable communication is possible from a to b, but is not possible from b to a. This result is somewhat counter-intuitive as the edges are bi-directional.

Organization. In Section 2, we describe our model, supply results from [1], and describe a simplified protocol SIMPLESEND which is analyzed in this paper. In Section 3, we study the properties of the "effective communication graph." In Section 4 we use these properties to prove that our protocol is efficient. In Section 5 we give a simple characterization of the possibility of reliable communication with one Byzantine processor. In Section 6, we prove that the simple necessary condition is not sufficient for $t \geq 2$, and reliable communication is not symmetric for $t \geq 2$.

2. PRELIMINARIES

2.1 The Model

The network is modeled by an undirected graph $G_C = \langle V, E_C \rangle$, where V is the set of parties in the network (i.e., $|V| = n$), and E_C describes the communication channels. That is, there is an edge $\langle u, v \rangle$ in E_C if and only if there is a communication channel between u and v. We assume that these communication channels are reliable: an adversary that does not control u or v (but might control all other vertices in the network) cannot change or delete a message sent on the edge $\langle u, v \rangle$ or insert a message on the channel. We assume that some pairs of parties share authentication keys. We informally explain what authentication schemes are; the reader is referred to, e.g., [15] for more details. An authentication scheme enables a sender and a receiver who share a common key to exchange messages such that the receiver can verify that the message was sent by the sender. We describe which pairs of parties have a common authentication key by a graph $G_A = \langle V, E_A \rangle$. That is, u and v have a common key, denoted by $k_{u,v}$, if and only if $\langle u, v \rangle \in E_A$. These keys are chosen according to some known probability distribution, and every set of vertices has no information on the keys of disjoint edges (except for their a-priori probability distribution).

We consider protocols for message transmission, in which a transmitter $a \in V$ wants to transmit a message M to a receiver $b \in V$. We assume that the system is synchronous. That is, a protocol proceeds in rounds; at the beginning of each round each party $v \in V$ sends messages to some of its neighbors in the graph G_C. These messages get to the neighbors before the beginning of the next round. We assume that all parties in the system know the topology of the graphs G_C and G_A. Furthermore, all the parties in the system know in which round party a starts to transmit a message to party b. The round complexity of a protocol is the number of rounds that have elapsed from its activation to its termination. The message complexity of a protocol is the total number of bits in messages exchanged in a round, maximized over all the rounds.

During the execution there might be Byzantine attacks (also known as "active attacks"). An adversary, with an unlimited power, controls a subset T of the parties. The adversary knows the protocol, the distribution under which the authentication keys where chosen, and the topology of the network (i.e., G_C and G_A). The adversary can choose T during the execution of the protocol. For every party in T, the adversary knows all the messages received by that party, its random inputs, and its keys. From the moment a party is included into T, the adversary determines the messages this party sends thereafter (possibly deviating from the protocol specification in an arbitrary manner).

DEFINITION 2.1 (RELIABLE PROTOCOL). *Let $a, b \in V$ be a transmitter and a receiver, and $t \leq n-2$. We say that a message transmission protocol from a to b is* t-reliable *if for every integer c there exists k_c such that for every message M of length at least k_c, when the adversary can control any set T of at most t parties such that $T \subseteq V \setminus \{a, b\}$, the probability that b accepts the message M, given that a transmitted M, is at least $1 - \frac{1}{|M|^c}$, where the probability is over the random inputs of the parties, the distribution of the authentication keys, and the random input of the adversary.*

In this paper we consider the problem of fault restricted reliable communication, which is a tool for characterizing when t-reliable transmission between a given pair of parties is possible. In the fault restricted version one of two given sets T_0, T_1, which are not necessarily disjoint, is guaranteed to contain all of the faulty processors.

DEFINITION 2.2 (FAULT RESTRICTED PROTOCOL). *Let a, b be a transmitter and a receiver, and let $T_0, T_1 \subseteq V \setminus \{a, b\}$. We say that a message transmission protocol from a to b is (T_0, T_1)-*reliable *if for every integer c there exists k_c such that for every message M of length at least k_c, when the adversary can control one of T_0, T_1, the probability that b accepts the message M, given that a transmitted M, is at least $1 - \frac{1}{|M|^c}$, where the probability is over the random inputs of the parties, the distribution of the authentication keys, and the random input of the adversary.*

It was shown in [1] that if there is a (T_0, T_1)-reliable protocol for every pair of sets of size at most t, then there is a t-reliable protocol. This t-reliable protocol executes (in parallel) the (T_0, T_1)-reliable protocol for every pair of sets of size t, and the receiver learns the message that was sent from the sender by analyzing the results of these executions. In particular, if t is constant and the (T_0, T_1)-reliable protocol is efficient for every T_0, T_1 of size at most t, then the resulting t-reliable protocol is efficient.

The reliability of a network is closely related to its connectivity. We consider *vertex* connectivity of *undirected* graphs. Two paths from a to b are *vertex disjoint* if no vertices other than a and b appear on both paths. A path P *passes* through a set T if there is a vertex $u \in T$ in the path. Otherwise, we say that P *misses* T. A graph $H = \langle V, E \rangle$ is (t, u, v)-*connected* if $\langle u, v \rangle \in E$ or if there are t vertex disjoint paths from u to v. There is an efficient algorithm that checks whether a graph is (t, u, v)-connected (see, e.g., [7]).

2.2 Characterizing Reliable Communication

In this section we quote the definition of G^* and a confusing pair from [1]. These definitions characterize when a can reliably communicate with b.

DEFINITION 2.3 (HONEST AND SEMI-HONEST PATHS). *Let $H = \langle V, E \rangle$ be a graph, u and v be some vertices in V, and T_0, T_1*

be subsets of V. A path $\langle u,\dots,v\rangle$ from u to v is honest *if it misses $T_0 \cup T_1$. A path $\langle u,\dots,v\rangle$ from u to v is* semi-honest *if it misses at least one of the sets T_0,T_1.*

To motivate the next definition consider an authentication edge $\langle u,v\rangle$ with a semi-honest path from u to v in G_C that passes through T_i, and a honest path from v to b in G_C. When u wants to send a message M to v, it authenticates M using the shared key $k_{u,v}$ and then sends the authenticated message along the semi-honest path from u to v. If the message never arrives at v or if it arrives with improper authentication, then v immediately knows that the set T_i is controlled by the adversary. Furthermore, v can share this information with b using the honest path from v to b.

DEFINITION 2.4 (THE GRAPH G^*). *Let $a,b \in V$ be the transmitter and the receiver, and $T_0,T_1 \subseteq V \setminus \{a,b\}$ be a pair of sets. Let $G_C = \langle V,E_C\rangle$ be the communication graph, and $G_A = \langle V,E_A\rangle$ be the authentication graph. Define $G_0 = \langle V,E_0\rangle$ where $E_0 = E_C$, and for every $j \geq 1$ define $G_j = \langle V,E_j\rangle$, where E_j is the union of E_{j-1} with the set of all authentication edges $\langle u,v\rangle \in E_A$ for which all of the following properties hold:*

1. *$u,v \notin T_0 \cup T_1$,*
2. *There is a semi-honest path from u to v in $G_{j-1} = \langle V,E_{j-1}\rangle$, and*
3. *There is a honest path in G_{j-1} from either u or v to b.*

Finally, define $G^* = G_n$.

Informally, the graph G^* is the "effective" communication graph, as it contains exactly the edges that can be used to reliably transmit a message from a to b. Property (2) ensures that v learns the Byzantine set if an improper message arrives from u, and Property (3) ensures that it can tell b about it. Also, as E_A is finite, there is a k for which $E_{k+i} = E_k$ for every $i \geq 0$. The graph G^* is defined as G_n since it is proven in [1] that $E_{n+i} = E_n$ for all $i \geq 0$.

REMARK 2.5. Authenticating a message M over an authentication edge $e = \langle u,v\rangle \in E_A$ is not necessary if there is a honest path from u to v in G_C. In such case, M is reliably transmitted over that path, and e can be discarded. Hence, w.l.o.g., we assume throughout the paper that there are no such edges in E_A.

We next define the notion of level of an edge, which is the stage in which it joins G^*. Formally, for an edge $e = \langle u,v\rangle$ define $\text{level}(e) \stackrel{\text{def}}{=} \min\{j|e \in E_j\}$. Note that e is a communication edge iff it has level 0. The level of a path P is defined by $\text{level}(P) = \max\{\text{level}(e)|e \in P\}$. Obviously, a path has level 0 iff it is a path in G_C. Also, for every authentication edge e with $\text{level}(e) = j$, there is a semi-honest path from u to v of level at most $j-1$, and there is a honest path from either u or v to b of level at most $j-1$. Therefore, if there is a honest path $P_{v,b}$ from v to b of level at most $j-1$, then there is another honest path $P_{u,b} = \langle u,v\rangle, P_{v,b}$ from u to b of level at most j. We conclude that there is a honest path from both u and v to b of level at most j.

We use $Graph1$ described in Fig. 1 to demonstrate these definitions. In this graph we have $\langle v,b\rangle \in E_1$ since $\langle v,t_1,b\rangle$ is a semi-honest path from v to b in G_0. Hence, the level of $\langle v,b\rangle$ is 1. Next, $\langle u,v\rangle$ is added to E_2 because $\langle u,t_0,b,v\rangle$ is a semi-honest path from u to b in G_1 and $\langle v,b\rangle$ is a honest path from v to b in G_1. Hence, the level of $\langle u,v\rangle$ is 2. Finally, the edge $\langle a,u\rangle$ is added to E_3 and its level is 3.

DEFINITION 2.6 (CONFUSING PAIR). *A pair (T_0,T_1) is an (a,b) confusing pair if $T_0,T_1 \subseteq V \setminus \{a,b\}$, and at least one of the following holds:*
1. *There is an index $i \in \{0,1\}$ such that every path from u to b in G_C passes through T_i.*
2. *Every path from a to b in G^* passes through $T_0 \cup T_1$.*

THEOREM 2.7 ([1]). *For all $T_0,T_1 \subseteq V \setminus \{a,b\}$ it holds that (T_0,T_1)-reliable message transmission from a to b is possible if and only if (T_0,T_1) is not an (a,b) confusing pair.*

2.3 The Depth of Edges

Beimel and Franklin used the level of edges in order to bound the round complexity of the protocol. The contribution of this paper is a more efficient protocol, and it starts with the introduction of depth of an edge. The depth of an edge is at most the level of an edge, but it can be significantly smaller. Moreover, the level of edges can be as much as $O(n)$ whereas the depth on an edge can be at most t.

We intuitively explain the following definition of the notion of depth. Consider some $t \in T_0 \cup T_1$. We say that a level $j+1$ is significant for t if j is the smallest for which t is in the connected component of b in G_j. The depth of an edge of level j is the number of levels $j' \leq j$ that are significant for some $t \in T_0 \cup T_1$.

DEFINITION 2.8 (DEPTH OF AN EDGE). *The following inductive definition over the graphs G_j is of subsets of $T_0 \cup T_1$. For $G_0 = G_C$ let $B_0 = \emptyset$ and for every $j \geq 1$, define B_j to be the set of all $t \in T_0 \cup T_1$, for which the following properties hold:*

- *For every $0 \leq j' < j$ it holds that $t \notin B_{j'}$, and*
- *For the $i \in \{0,1\}$ such that $t \in T_i$ there is a path from t to b in G_{j-1} that misses $T_{\bar{i}}$.*

We denote depth$(j) = |\{j'|B_{j'} \neq \emptyset, 1 \leq j' \leq j)\}|$, and say that an edge e is of depth d if depth(level$(e)) = d$.

Note that e is of depth 0 iff $e \in G_C$ iff e is of level 0. For a path P we define $\text{depth}(P) = \max\{\text{depth}(e)|e \in P\}$. Therefore, a path P is of depth 0 iff P is in G_C iff the level of P is 0. The depth of the graph G^* is the maximal depth over all the edges in G^*. For example, in $Graph2$ described in Fig. 1 we have $B_1 = \{t_0,t_1\}$. Hence, all of the authentication edges are of depth 1 and the depth of $Graph2$ is 1. We next bound the depth of G^*.

LEMMA 2.9. *If there is no honest path from a to b in G_C and G_C is $(t+1,a,b)$-connected, then the depth of G^* is at most t.*

PROOF. Let G_C be the communication graph. Since G_C is $(t+1,a,b)$-connected, there are at least $t+1$ disjoint paths from a to b in G_C. If there is no honest path from a to b in G_C, then none of these paths is honest and there is at least one Byzantine vertex on each one of them. From each of these Byzantine vertices there is a path to b that has no other Byzantine vertices on it, and therefore $|B_1| \geq t+1$. Thus, there are at most another $2t-(t+1) = t-1$ sets B_j for which $B_j \neq \emptyset$, and the depth of G^* is as asserted. $\square$

2.4 The Protocol SimpleSend

The procedure SIMPLESEND(M,u,v), described in Fig. 2, transmits a message through a path in G^*. For every authentication edge $\langle u',v'\rangle$ on the path it recursively calls to SIMPLESEND(M,u',v') to transmit the message with its authentication on a path from u to v. This procedure guarantees that if the original path contains no Byzantine processors and the message arrives at v then this message is indeed the message that u sent. However, if these conditions do not hold then no guarantees are made. The protocol

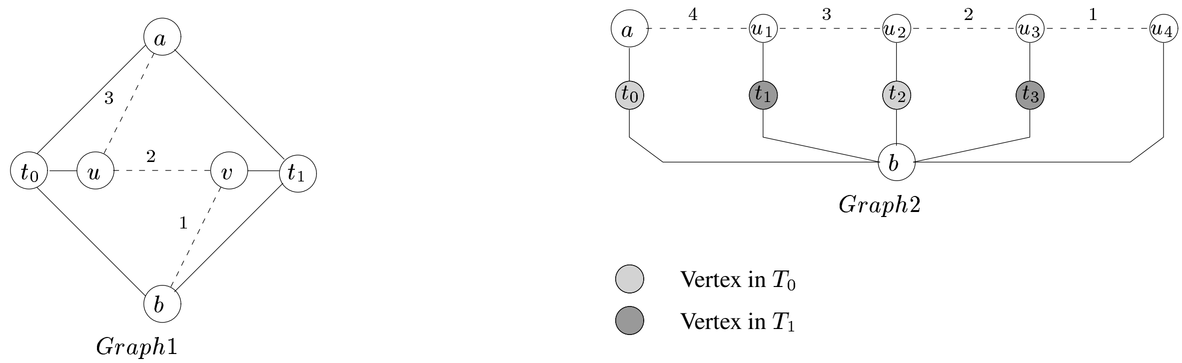

Figure 1: Examples of partially authenticated networks. The numbers indicate the level of authentication edges.

SIMPLESEND(M, u, v) is a simplified version of the protocol SEND(M, u, v) from [1]. The two protocols propagate the message M to b on the same paths using the same recursive calls. However, Protocol SEND(M, u, v) has an additional alert mechanism, which means that every v that should have gotten an authenticated message from u sends, in parallel to the "main execution," a message to b notifying b if it got the message or not. In [1] it is explained how this information enables b to accept the correct message.

EXAMPLE 2.10. Consider $Graph1$ described in Fig. 1. We can use the following protocol to $(\{t_0\}, \{t_1\})$-reliably transmit a message M from a to b: M is sent over the semi-honest paths $\langle a, t_0, b\rangle$ and $\langle a, t_1, b\rangle$ and on the honest path $\langle a, u, v, b\rangle$. To transmit M over $\langle a, u, v, b\rangle$ the recursive protocol SIMPLESEND(M, a, b) is executed. If M arrives on $\langle a, u, v, b\rangle$ then b accepts it. Otherwise, at least one authentication edge was disabled and b is informed by the alert mechanism which of t_0, t_1 is Byzantine. Since b knows which of the paths $\langle a, t_0, b\rangle$, $\langle a, t_1, b\rangle$ is Byzantine free, it can choose the message delivered on this path.

In this work we only analyze the round complexity of the protocol SIMPLESEND(M, u, v). Claim 2.11, which is implicit in [1], proves that SEND(M, u, v) is efficient if SIMPLESEND(M, u, v) is efficient.

CLAIM 2.11. *If for every $u \in V$ protocol* SIMPLESEND(M, u, b) *terminates after at most τ rounds, then for every $u \in V$ protocol* SEND(M, u, b) *terminates after at most $n \cdot \tau$ rounds.*

Protocol SIMPLESEND(M, u, v) can choose any semi-honest path from u to v. Exploiting the special structure of G^*, we show in the rest of the paper how to choose these paths such that the resulting protocol is efficient (at least for a constant number of Byzantine processors). As observed in [1], for every authentication edge $\langle u, v\rangle$ of level j there is a path from u to v in G^* of level $j - 1$. Thus, transmitting a message on an authentication edge of level j can be done by at most n transmissions on edges of level $j - 1$, yielding a protocol with round complexity $n^{O(n)}$.

The first property that we introduce is of paths that end in b. Specifically, for every authentication edge $\langle u, v\rangle$ there is a path from both u and v to b which has at most one edge of each level. The concatenation of the path from u to b with the path from b to v is a path from u to v that has at most two edges of each level. By simple induction this yields a protocol with round complexity $2^{O(n)}$.

Both approaches fail to consider the impact of the number of Byzantine vertices on the round complexity of the protocol. The main contribution of this paper is the concept of depth. When we send a message from u to b we choose a path in which the depths of authentication edges do not increase. We prove an upper bound on the round complexity of sending a message over an authentication edge that is exponential in the depth of the edge and linear in its level. The resulting protocol has round complexity $n^{O(t)}$.

EXAMPLE 2.12. Consider $Graph2$ described in Fig. 1 in which $T_0 = \{t_0, t_2\}$ and $T_1 = \{t_1, t_3\}$. To send a message over the authentication edge $\langle a, u_1\rangle$, the semi-honest path $\langle a, t_0, b, t_2, u_2, u_1\rangle$ can be used. This requires a recursive send on the authentication edge $\langle u_1, u_2\rangle$. To send a message over $\langle u_1, u_2\rangle$ we can use the semi-honest path $\langle u_1, t_1, b, t_3, u_3, u_2\rangle$ which requires a recursive send on the authentication edge $\langle u_2, u_3\rangle$. For the edge $\langle u_2, u_3\rangle$ we use the semi-honest path $\langle u_2, t_2, b, u_4, u_3\rangle$ which requires a recursive send on the authentication edge $\langle u_3, u_4\rangle$.

Artificial example as it may seem, we show in Lemma 4.1 that every graph has the structure of $Graph2$ and then we analyze the transmission costs in such structure. We show that these costs are linear with respect to the level and exponential with respect to the depth. The somewhat technical proofs in Section 3 provide us with the tools that enable the construction of such structure.

The following lemma, which is used in Section 4, proves that the round complexity of transmitting M from u to v is equal to the round complexity of transmitting M from v to u for all $u, v \in V$. This implies that the round complexity of the protocol could be analyzed regardless of the direction upon which M is sent.

LEMMA 2.13. *If there is an implementation of the protocol* SIMPLESEND(M, u, v) *that terminates after ℓ rounds, then there is an implementation of* SIMPLESEND(M, v, u) *that terminates after ℓ rounds.*

PROOF. First, assume that recursive calls to SIMPLESEND terminate after only one round. Let $P_{u,v}$ be the semi-honest path chosen in the execution of SIMPLESEND(M, u, v). Since the reverse path $P_{v,u}$ is a semi-honest path as well, and since we assume that recursive calls to SIMPLESEND terminate after one round, the round complexity of SIMPLESEND(M, u, v) is equal to the round complexity of SIMPLESEND(M, v, u). It is possible to avoid the assumption that recursive calls to SIMPLESEND terminate after one round by using induction. $\square$

The fact that Protocol SIMPLESEND is symmetric with respect to the sender and the receiver does not imply that reliable communication is symmetric with respect to the sender and the receiver. The reason is that the alert mechanism added to SIMPLESEND is not symmetric.

PROTOCOL SIMPLESEND(M,u,v)

PARAMETERS: M - message, u - source, v - target.

Choose a semi-honest path $v_0,\ldots,v_\ell$ from u to v in G^*.

FOR $i=0$ TO $\ell-1$ DO (* v_i propagates the message to v_{i+1} *)

IF $\langle v_i,v_{i+1}\rangle \in E_C$ THEN v_i sends M to v_{i+1} on this edge

OTHERWISE, $\langle v_i,v_{i+1}\rangle \in E_A$:

1. v_i executes

   SIMPLESEND$(\langle M, \text{AUTH}(M,k_{v_i,v_{i+1}})\rangle, v_i, v_{i+1})$

2. IF v_{i+1} received $\langle \hat{M},\hat{\alpha}\rangle$ such that $\hat{\alpha} \neq \text{AUTH}(\hat{M},k_{v_i,v_{i+1}})$ THEN reject

Figure 2: A protocol for sending a message from u to v.

3. PROPERTIES OF THE GRAPH G^*

In this section we analyze the graph G^*. In particular, we show that paths that end in b have additional properties. Our protocol utilizes this analysis in order to more effectively transmit a message over an authentication edge.

3.1 Monotonicity

The first property that we introduce is path monotonicity. Specifically, monotonous paths have only one authentication edge of each level. As explained above, monotonous paths imply a protocol with round complexity $2^{O(n)}$.

DEFINITION 3.1 (MONOTONOUS PATH). *A path P is* monotonous *if for all authentication edges e_1 and e_2 in P, whenever e_2 precedes e_1 in the path P, then level(e_2) is strictly larger then level(e_1).*

For example, the path $\langle a,u_1,u_2,u_3,u_4,b\rangle$ in $Graph2$ (described in Fig. 1) is a monotonous path. Note that P is monotonous implies that the first authentication edge e on P has the highest level over all of the other edges in P. Hence, the level of P is determined by the level of this edge and vice versa. Also, note that if P is of level 0 (i.e., P is a path in G_C), then P is monotonous.

LEMMA 3.2. *For every $w\in V$, if there is a honest path from w to b in G^* of level j, then there is a monotonous honest path from w to b of level at most j.*

PROOF. The lemma is proved by induction on j. The base case for $j=0$ follows from the observation that every path of level 0 is monotonous. For the induction step, assume that for every $w\in V$, if there is a honest path from w to b of level at most j, then there is a monotonous honest path from w to b of level at most j. Now, let $P_{w,b}$ be a honest path from w to b of level $j+1$. Since the level of $P_{w,b}$ is at least 1, there is at least one authentication edge on $P_{w,b}$. Denote the first authentication edge on $P_{w,b}$ by $e=\langle u,v\rangle$. If there is a honest path $P_{u,b}$ from u to b of level at most j, then concatenating the prefix $\langle w,\ldots,u\rangle$ of $P_{w,b}$ with $P_{u,b}$ yields a honest path $\langle w,\ldots,u\rangle, P_{u,b}$ from w to b of level at most j, and by the induction hypothesis there is a monotonous honest path from w to b of level at most j. Otherwise, by Property (3) in the definition of the graph G_j, the level of e must be exactly $j+1$ and there is a honest path from v to b with level at most j. By the induction hypothesis there is a monotonous honest path $P_{v,b}$ from v to b of level at most j, and the path $\langle w,\ldots,u\rangle,\langle u,v\rangle,P_{v,b}$ is a monotonous honest path from v to b of level $j+1$, and the induction follows. □

3.2 Left Edges and Left Paths

We further introduce the second property of paths that end in b, which we call left paths.

DEFINITION 3.3 (LEFT AND RIGHT EDGES). *An authentication edge $e=\langle u,v\rangle$ of level j is* left *if the following properties hold:*

1. *There is a honest path from v to b of level at most $j-1$.*
2. *There is a semi-honest path $P_{u,v}$ from u to v of level at most $j-1$, with at least one Byzantine vertex on this path, where for the leftmost Byzantine vertex t on $P_{u,v}$, the prefix $\langle u,\ldots,t\rangle$ of $P_{u,v}$ is in G_C.*

An edge $\langle u,v\rangle$ is right iff $\langle v,u\rangle$ is left. A path P is left if e is left for every authentication edge $e\in P$.

For an illustration of a left edge see Fig. 4 case (1). For example, the authentication edge $\langle a,u_1\rangle$ of level 4 in $Graph2$ described in Fig. 1 is left since $\langle a,t_0,b,t_2,u_2,u_1\rangle$ is a semi-honest path from a to u_1 with t_0 as its leftmost Byzantine vertex and $\langle u_1,u_2,u_3,u_4,b\rangle$ is honest path from u_1 to b of level 3. Definition 2.4 of the graph G^* implies that there must be a honest path from either u or v to b of level at most $j-1$, and a semi-honest path $P_{u,v}$ from u to v of level at most $j-1$. Property 2 in Definition 3.3 requires, in addition, that a Byzantine vertex must appear on $P_{u,v}$ before any authentication edges that are on $P_{u,v}$. Informally, this vertex provides a shortcut path to b that enables sending messages more efficiently.

LEMMA 3.4. *Every authentication edge in G^* is either left or right.*

PROOF. Let $e=\langle u,v\rangle$ be an authentication edge of level j. We prove by induction on j, that e is either left or right. For every edge of level 1 there is a semi-honest path from u to v in G_C. Remark 2.5 implies that there must be at least one Byzantine vertex on this path. If there is a honest path from v to b of level 0, then e is left. Otherwise, there is a honest path from u to b of level 0 and e is right.

Assume that every authentication edge of level at most $j-1$ is either left or right. The induction step for j is as follows: Let $e=\langle u,v\rangle$ be an edge of level j. If there is a semi-honest path from u to v in G_C, then similar arguments to those in the base case hold, and e is either left or right. Otherwise, let P be a semi-honest path from u to v with at least one authentication edge, and choose P with a minimal level among the semi-honest paths from u to v. Denote the level of P by j', where $1\le j' < j$, and let $e_1=\langle u_1,v_1\rangle$ and $e_2=\langle u_2,v_2\rangle$ be the leftmost and rightmost authentication edges on P, respectively (e_1 and e_2 can be the same edge). Denote $P_{u_1,b}$ and $P_{v_2,b}$ to be honest minimal level paths from u_1 and v_2, respectively, to b. Define $P_{u,v} \stackrel{\text{def}}{=} \langle u,\ldots,u_1\rangle, P_{u_1,b}, P_{b,v_2}, \langle v_2,\ldots,v\rangle$. Note that $P_{u,v}$ is a semi-honest path from u to v of level at most j' that misses $T_{\bar{i}}$ for some $i\in\{0,1\}$. Since $P_{u_1,b},P_{b,v_2}$ is a honest path, any Byzantine vertex on $P_{u,v}$, if there is any, may appear only on $\langle u,\ldots,u_1\rangle$ or $\langle v_2,\ldots,v\rangle$. There are three cases to consider; in each case we construct the paths proving that e is either left or right.

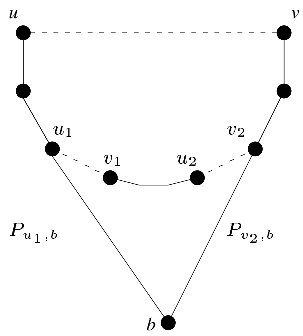

Figure 3: Case 3 in the proof of Lemma 3.4.

- There are $t_1, t_2 \in T_i$ such that t_1 is a Byzantine vertex in $\langle u, \ldots, u_1\rangle$, and t_2 is the a Byzantine in $\langle v_2, \ldots, v\rangle$: Note that there is $w \in \{u, v\}$ for which there is a honest path from w to b of level at most $j - 1$. If $w = u$ then e is right. Otherwise, $w = v$ and e is left. See Fig. 4 case (1).
- There is $t_1 \in T_i$ such that t_1 is a Byzantine vertex in $\langle u, \ldots, u_1\rangle$, and $\langle v_2, \ldots, v\rangle$ misses $T_0 \cup T_1$: In this case the prefix $\langle u, \ldots, t_1\rangle$ of $P_{u,v}$ is in G_C. Also, the honest paths $\langle v, \ldots, v_2\rangle$ and $P_{v_2,b}$ make a honest path $\langle v, \ldots, v_2\rangle, P_{v_2,b}$ from v to b of level at most $j - 1$, which implies that e is left. See Fig. 4 case (2). The case where there is a Byzantine vertex in $\langle v_2, \ldots, v\rangle$, and $\langle u, \ldots, u_1\rangle$ misses $T_0 \cup T_1$ is symmetric.
- Both $\langle u, \ldots, u_1\rangle$, and $\langle v_2, \ldots, v\rangle$ miss $T_0 \cup T_1$: By the induction hypothesis, each of e_1 and e_2 is either left or right. If e_1 is right and e_2 is left, then, by Definition 3.3, the level of $P_{u_1,b}, P_{b,v_2}$ is at most $j' - 1$. See Fig. 3. This implies that $P_{u,v}$ is a semi-honest path from u to v of level at most $j' - 1$, contradiction to the choice of P with a minimal level. Hence, either e_1 is left or e_2 is right. If e_1 is left, then by the induction hypothesis there is a semi-honest path P_{u_1,v_1} from u_1 to v_1 of level at most $j' - 1$, and there is a prefix $\langle u_1, \ldots, t_1\rangle$ of P_{u_1,v_1} where $t_1 \in T_i$ is the leftmost Byzantine on P_{u_1,v_1} for some $i \in \{0, 1\}$. Note that $\langle u_1, \ldots, t_1\rangle$ is a path in G_C. We construct a semi-honest path P' from $P_{u,v}$ by replacing e_1 with P_{u_1,v_1}. See Fig. 4 case (3). There is a prefix of P' in which t_1 is the leftmost Byzantine. Moreover, the level of P' is at most j'. Finally, since $\langle v, \ldots, v_2\rangle, P_{v_2,b}$ is a honest path from v to b of level at most $j - 1$, then e is left. If e_2 is right, then by symmetric arguments e is right.

Thus, the induction follows. $\square$

The next lemma combines the property of monotonicity with the property of left paths. Our protocol uses both the monotonicity of paths and their left structure to transmit messages efficiently.

LEMMA 3.5. *For every left authentication edge $\langle u, v\rangle$ of level j, there is a* left, *monotonous, honest path from v to b of level at most $j - 1$.*

PROOF. We prove by induction on j that for every left authentication edge $e = \langle u, v\rangle$ of level j there is a left, monotonous, honest path from v to b of level at most $j - 1$. Let $e = \langle u, v\rangle$ be a left authentication edge of level $j \geq 1$. For the base case of the induction, the level of e is 1 and e is left. By Definition 3.3 there is a honest path from v to b of level 0. Since this path is in G_C it is left and monotonous as well.

Assume that the induction hypothesis holds for every authentication edge e of level at most j. For the induction step, let $e = \langle u, v\rangle$ be a left authentication edge of level $j + 1$. By Definition 3.3 there is a honest path from v to b of level at most j. Therefore, there is a minimal $j' \leq j$ for which there is a honest path from v to b of level j'. By Lemma 3.2, there is a monotonous, honest path $P_{v,b}$ from v to b of level j'. We show that there is a left, monotonous, honest path from v to b of level j'. If $j' = 0$, then, by Definitions 3.3 and 3.1, $P_{v,b}$ is a left, monotonous, honest path from v to b. Otherwise, there is a leftmost authentication edge $\langle u', v'\rangle$ on $P_{v,b}$ of level at most j', and by Lemma 3.4, the edge $\langle u', v'\rangle$ is either left or right. If $\langle u', v'\rangle$ is right then there is a honest path $P_{u',b}$ from u' to b of level at most $j' - 1$, and $\langle v, \ldots, u'\rangle, P_{u',b}$ is a honest path from v to b of level at most $j' - 1$, contradiction to the choice of $P_{v,b}$ with a minimal level. Therefore, $\langle u', v'\rangle$ is a left edge and level$(\langle u', v'\rangle) \leq j'$. By the induction hypothesis, there is a left, monotonous, honest path $P_{v',b}$ from v' to b of level at most level$(\langle u', v'\rangle) - 1$. Therefore, $\langle v, \ldots, u'\rangle, \langle u', v'\rangle, P_{v',b}$ is a left, monotonous, honest path from v to b of level at most j, as asserted. $\square$

In the next lemma we make the first link between depth and left edges.

LEMMA 3.6. *For every left authentication edge $e = \langle u, v\rangle$ of depth d there is a semi-honest path from u to b of depth $\leq d - 1$.*

PROOF. Let $e = \langle u, v\rangle$ be a left authentication edge of level j and depth d. Since e is left, there is a semi-honest path $P_{u,v}$ from u to v of level at most $j - 1$ and there is a honest path $P_{v,b}$ from v to b of level at most $j - 1$. Hence, the path $P_{u,v}, P_{v,b}$ is a semi-honest path from u to b of level at most $j - 1$ and there is a leftmost Byzantine vertex $t \in T_i$ on that path for some $i \in \{0, 1\}$. This implies that there is also a semi-honest path from t to b of level at most $j - 1$ and therefore $t \in B_k$ for some $k \leq j$. Note that the prefix $\langle u, \ldots, t\rangle$ of $P_{u,v}$ misses $T_{\bar{i}}$. Since $t \in B_k$, there is a semi-honest path $P_{t,b}$ from t to b in G_{k-1} that misses $T_{\bar{i}}$. Also, $B_k \neq \emptyset$ implies that depth$(k-1) = depth(k) - 1 \leq depth(j) - 1 = d - 1$, and we conclude that $\langle u, \ldots, t\rangle, P_{t,b}$ is a semi-honest path from u to b of depth at most $d - 1$. $\square$

4. EFFICIENT IMPLEMENTATION OF PROTOCOL SIMPLESEND

In this section we consider the depth of paths used by the protocol, in order to better analyze its running time. We express the transmission cost in terms of depth, which is at most t, and prove that the running time of the protocol is $n^{O(t)}$. In particular, this implies that the protocol is efficient whenever the number of Byzantine vertices is constant.

We use the following notation throughout our analysis: For all vertices $w \in V$ such that there is semi-honest path from w to b of depth at most d define cost(d) to be an upper bound on the running time of of SIMPLESEND(M, w, b), taken as the minimal over all of the implementations of SIMPLESEND(M, w, b). Since a path from w to b with depth 0 can have at most n edges, all of which are communication edges, we conclude that cost$(0) \leq n$. The intuition behind the definition of cost(d) is that we can use paths of depth at most $d - 1$ to send a message over an authentication edge of depth d. This enables us to express cost(d) in terms of cost$(d - 1)$.

LEMMA 4.1. *Let $e = \langle u, v\rangle$ be a left authentication edge of depth d, and let $P_{v,b}$ be a left, monotonous, honest path from v to b with at most m authentication edges of depth d. Then there is an implementation of* SIMPLESEND(M, u, v) *that terminates after at most* $2(m + 1) \cdot cost(d - 1) + mn$ *rounds.*

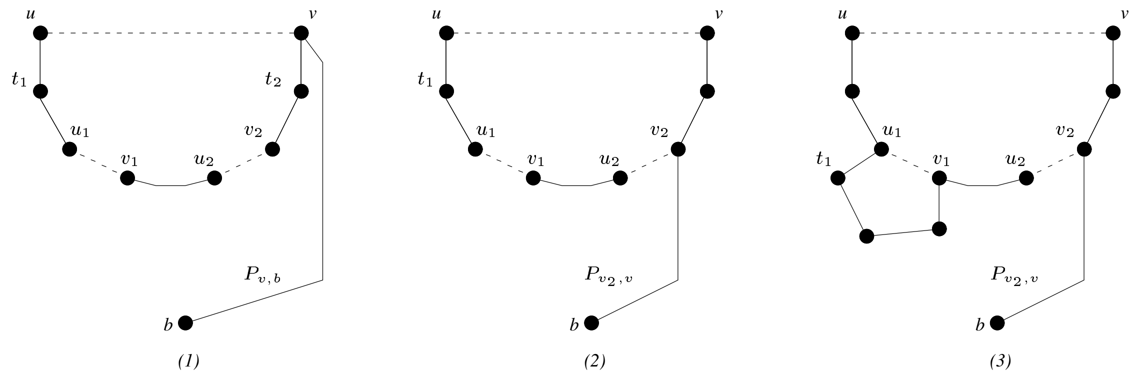

Figure 4: The three cases in the proof of Lemma 3.4.

PROOF. Since e is left, then by Lemma 3.6 there is a semi-honest path $P_{u,b}$ from u to b of depth at most $d-1$. By the definition of $\text{cost}(d-1)$ it holds that a message M sent from u to b by $\text{SIMPLESEND}(M,u,b)$ arrives at b after at most $\text{cost}(d-1)$ rounds. By induction on m, which is the number of authentication edges of depth d on $P_{v,b}$, we prove that $\text{SIMPLESEND}(M,u,v)$ terminates after at most $2(m+1)\cdot\text{cost}(d-1)+mn$ rounds. For the base case, since $m=0$ then the honest path $P_{v,b}$ is of depth at most $d-1$. Lemma 2.13 guarantees that $\text{SIMPLESEND}(M,b,v)$ requires the same number of rounds as $\text{SIMPLESEND}(M,v,b)$. Hence, a message M sent from b to v by $\text{SIMPLESEND}(M,b,v)$ arrives at v after at most $\text{cost}(d-1)$ rounds. Therefore, the path $P_{u,b},P_{b,v}$ is a semi-honest path of depth at most $d-1$, and a message M sent by $\text{SIMPLESEND}(M,u,v)$ from u to v through b, arrives at v after at most $2\cdot\text{cost}(d-1)$ rounds.

Assume the induction hypothesis holds for every $m'\leq m$. For the induction step, let $e=\langle u,v\rangle$ be a left edge of depth d, and fix $P_{v,b}$ to be a left, monotonous, honest path from v to b with $m+1$ authentication edges of depth d. Denote $P_{v,b}\stackrel{\text{def}}{=}P_{v,v_{m+1}},P_{v_{m+1},b}$ where $P_{v,v_{m+1}}=\langle v\rightsquigarrow u_1,v_1\rightsquigarrow u_2,v_2,\ldots,u_{m+1},v_{m+1}\rangle$ is a prefix of $P_{v,b}$ with $m+1$ authentication edges $e_\ell=\langle u_\ell,v_\ell\rangle$ for every $1\leq\ell\leq m+1$ (the notation $\rightsquigarrow$ stands for a honest path in G_C), and $P_{v_{m+1},b}$ is a suffix of $P_{v,b}$ of depth at most $d-1$.

Consider the path $P_{v,v_{m+1}}$. This path is also a left, monotonous, honest path from v to v_{m+1}. By Lemma 3.6 there is a semi-honest path $P_{u_\ell,b}$ from u_ℓ to b of depth at most $d-1$ for every $1\leq\ell\leq m+1$ (see Fig. 5). This implies that there is a semi-honest path $P_{b,v_\ell}=P_{b,u_{\ell+1}},\langle u_{\ell+1},\rightsquigarrow,v_\ell\rangle$ from b to v_ℓ of depth at most $d-1$ for every $1\leq\ell\leq m$, where $P_{b,u_{\ell+1}}$ is the reverse path of $P_{u_{\ell+1},b}$. Let T_i be the set missed by $P_{u,b}$. There are two cases:

First Case. For every $1\leq\ell\leq m+1$ the semi-honest path $P_{u_\ell,b}$ from u_ℓ to b misses $T_{\bar{i}}$: For every $1\leq\ell\leq m$ consider the path $P_{u_\ell,b},P_{b,v_\ell}$. This is a semi-honest path from u_ℓ to v_ℓ of level at most $d-1$. For the edge e_{m+1}, recall that $P_{v_{m+1},b}$ is a honest path from v_{m+1} to b of depth at most $d-1$. Hence, there is a semi-honest path $P_{u_{m+1},b},P_{b,v_{m+1}}$ from u_{m+1} to v_{m+1} of depth at most $d-1$. This implies that $\text{SIMPLESEND}(M,u_\ell,v_\ell)$ terminates after at most $2\cdot\text{cost}(d-1)$ rounds for every $1\leq\ell\leq m+1$.

Consider the semi-honest path from u to v:

$$P_{u,b},P_{b,v_{m+1}},\langle v_{m+1},u_{m+1},\ldots,v_1,u_1\rightsquigarrow v\rangle.$$

A message M sent from u on $P_{u,b},P_{b,v_{m+1}}$ arrives at v_{m+1} after at most $2\cdot\text{cost}(d-1)$ rounds. Since there are at most n communication edges on $P_{v,v_{m+1}}$, each with transmission cost of 1 round, a message M sent from u to v by $\text{SIMPLESEND}(M,u,v)$ arrives at v after at most $2\cdot\text{cost}(d-1)+(m+1)\cdot2\cdot\text{cost}(d-1)+n=2(m+2)\cdot\text{cost}(d-1)+n$ rounds.

Second Case. There is an ℓ, where $1\leq\ell\leq m+1$, for which the path $P_{u_\ell,b}$ passes through $T_{\bar{i}}$: Let m' be the minimal for which the semi-honest path $P_{u_{m'},b}$ passes through $T_{\bar{i}}$. Since the semi-honest path $P_{u_{m'},b}$ passes through $T_{\bar{i}}$, it misses T_i. Also, by the choice of m' the path $P_{u_\ell,b}$ misses $T_{\bar{i}}$ for every $1\leq\ell\leq m'-1$. As in the previous case the path $P_{u_\ell,b},P_{b,v_\ell}$ is a semi-honest path from u_ℓ to v_ℓ of level at most $d-1$ for every $1\leq\ell<m'-1$, and $\text{SIMPLESEND}(M,u_\ell,v_\ell)$ terminates after at most $2\cdot\text{cost}(d-1)$ rounds for every $1\leq\ell<m'-1$.

Consider the semi-honest path from u to v:

$$P_{u,b},P_{b,u_{m'}},\langle u_{m'}\rightsquigarrow v_{m'-1},u_{m'-1},\ldots,v_1,u_1\rightsquigarrow v\rangle.$$

By the induction hypothesis for the edge $\langle u_{m'-1},v_{m'-1}\rangle$ it holds that $\text{SIMPLESEND}(M,u_{m'-1},v_{m'-1})$ terminates after at most $2\cdot[(m+1)-m'+1]\cdot\text{cost}(d-1)+[(m+1)-m']n$ rounds. Since there are at most n communication edges on $P_{v,u_{m'}}$, each with transmission cost of 1 round, we conclude that a message M sent from u by $\text{SIMPLESEND}(M,u,v)$ arrives at v after at most $2\cdot\text{cost}(d-1)+(m'-1)\cdot2\cdot\text{cost}(d-1)+2[m-m']\cdot\text{cost}(d-1)+[m+1-m']n+n\leq2(m+2)\cdot\text{cost}(d-1)+(m+1)n$, and the induction follows. $\square$

The next lemma uses Lemma 4.1 to explicitly evaluate $\text{cost}(d)$. Towards this goal, define $\delta_0=0$ and $\delta_d=|\{j|\text{depth}(j)=d\}|$ for every $d\geq1$. That is, δ_d is the number of levels in which the depth of edges is d. Clearly, $\delta_0+\cdots+\delta_d\leq n$.

LEMMA 4.2. *For every depth $d\geq0$ it holds that: cost$(d)\leq(d+1)\cdot n\prod_{k=0}^{d}(\delta_k+1)^2$.*

PROOF. We prove by induction on d, that for every vertex $w\in V$ with a semi-honest path P from w to b of depth d the cost of sending a message from w to b is as promised. For the base case, any path of depth 0 is a path in G_C, which implies that $\text{cost}(0)\leq n$ and the inequality holds. Assume the induction hypothesis for every $d'<d$. For the induction step, let P be a semi-honest path from w to b of level j such that the level of P is minimal among the semi-honest paths from w to b, and the depth of P is d. Since $d\geq1$, there is at least one authentication edge on P. Let $\langle u,v\rangle$ be the leftmost authentication edge on P, and let $\langle w,\ldots,u\rangle$ be a prefix of P in G_C. If $\langle u,v\rangle$ is right, then there is a honest path P' from u to b of level at most $j-1$, which implies that $\langle w,\ldots,u\rangle,P'$ is a semi-honest path from w to b of level at most $j-1$, contradiction to the choice of P with a minimal level. Therefore $\langle u,v\rangle$ is left, and

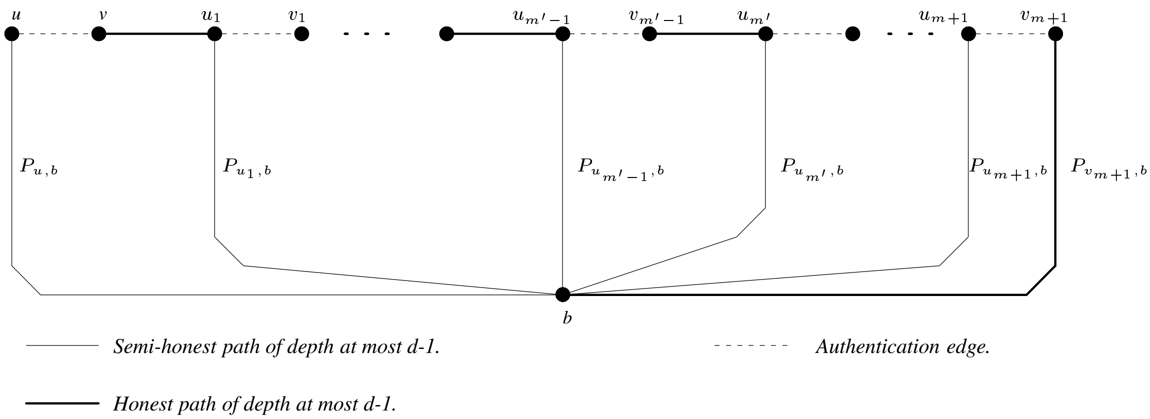

Figure 5: The paths in the induction step of the proof of Lemma 4.1.

by Lemma 3.5 we choose $P_{v,b}$ to be a left, monotonous, honest path from v to b of depth d and level at most $j-1$.

Consider the semi-honest path $P_{w,b} \stackrel{\text{def}}{=} \langle w, \ldots, u \rangle, \langle u, v \rangle, P_{v,b}$, and note that $\langle u, v \rangle, P_{v,b}$ is a left, monotonous, honest path from u to b of depth at most d. By the definition of δ_d and the monotonicity of $\langle u, v \rangle, P_{v,b}$ there are $m \leq \delta_d$ authentication edges of level d on $\langle u, v \rangle, P_{v,b}$. Let $\langle u_m, v_m \rangle = \langle u, v \rangle$ and define $P_{u,b} \stackrel{\text{def}}{=} \langle u_m, v_m, \ldots, u_1, v_1 \rangle, P_{v_1,b}$ where $e_\ell = \langle u_\ell, v_\ell \rangle$ is an authentication edge of depth d for every $1 \leq \ell \leq m$, and $P_{v_1,b}$ is a path from v_1 to b of depth at most $d-1$.

$P_{v_1,b}$ is a semi-honest path in particular, and thus a message M sent from v_1 to b by SIMPLESEND(M, v_1, b) arrives at b after at most cost$(d-1)$ rounds. In addition, by Lemma 4.1 a message M sent from u_ℓ to v_ℓ by SIMPLESEND(M, u_ℓ, v_ℓ) arrives at v_ℓ after at most $2 \cdot \ell \cdot \text{cost}(d-1) + (\ell-1)n$ rounds for every $m \geq \ell \geq 1$. Finally, since there are at most n communication edges on P_{w,v_1} we conclude that a message M sent from w by SIMPLESEND(M, w, b) arrives at b after at most $\sum_{\ell=1}^{m}[2 \cdot \ell \cdot \text{cost}(d-1) + (\ell-1)n] + \text{cost}(d-1) + n$ rounds, where $m \leq \delta_d$. Thus:

$$\begin{aligned}
&\text{cost}(d)\\
&\leq \sum_{\ell=1}^{m}[2 \cdot \ell \cdot \text{cost}(d-1) + (\ell-1)n] + \text{cost}(d-1) + n\\
&\leq \sum_{\ell=1}^{\delta_d} \ell[2 \cdot \text{cost}(d-1) + n] + \text{cost}(d-1) + n\\
&\leq \left(\frac{(\delta_d+1)\delta_d}{2} + 1\right)[2 \cdot \text{cost}(d-1) + n]\\
&\leq (\delta_d+1)^2[d \cdot n \prod_{k=0}^{d-1}(\delta_k+1)^2 + n] \qquad (1)\\
&\leq (d+1) \cdot n \prod_{k=0}^{d}(\delta_k+1)^2.
\end{aligned}$$

The inequality in (1) is implied by the induction hypothesis. ☐

LEMMA 4.3. *The round complexity of* SIMPLESEND(M, a, b) *is at most* $n^2 \cdot \left(\frac{2n}{t}\right)^{2t}$.

PROOF. By Lemma 2.9 the depth of G^* is at most t, which implies by Lemma 4.2 that SIMPLESEND(M, a, b) terminates after at most cost$(t) \leq (t+1) \cdot n \prod_{k=0}^{t}(\delta_k+1)^2$ rounds. Let j be the highest level of an edge in G^*, and notice that $\delta_0 + \delta_1 + \cdots + \delta_t = j$ and that $j \leq n$. Also, $\prod_{k=0}^{t}(\delta_k+1)^2$ is maximal when $\delta_1 = \delta_2 = \cdots = \delta_t = \frac{j}{t} \leq \frac{n}{t}$. Finally, since $t \leq n-2$, we conclude that:

$$\begin{aligned}
\text{cost}(t) &\leq (t+1)n \prod_{k=0}^{t}(\delta_k+1)^2 \leq n^2 \prod_{k=1}^{t}\left(\frac{n}{t}+1\right)^2\\
&= n^2\left(\frac{n+t}{t}\right)^{2t} \leq n^2\left(\frac{2n}{t}\right)^{2t}.
\end{aligned}$$

☐

THEOREM 4.4. *If t-reliable communication from a to b is possible, then there is a t-reliable protocol from a to b with round complexity at most $n^3 \cdot \left(\frac{2n}{t}\right)^{2t} \leq 2^{O(n)}$ rounds.*

PROOF. To achieve (T_0, T_1)-reliable communication from a to b we execute the protocol SEND(M, a, b) of [1]. By Claim 2.11 the round complexity of this protocol is at most n times the round complexity of SIMPLESEND(M, a, b), due to alert messages overhead. We next follow the technique used in [1], and execute SEND(M, a, b) for every pair $T_0, T_1 \subseteq V \setminus \{a, b\}$ with $|T_0|, |T_1| = t$. There are $\binom{n}{t}^2$ such executions, and we let them run in parallel. By Lemma 4.3 we conclude that the round complexity of SEND(M, a, b) is at most $n^3 \cdot \left(\frac{2n}{t}\right)^{2t} \leq 2^{O(n)}$ rounds. ☐

COROLLARY 4.5. *For every constant t, if t-reliable communication from a to b is possible, then there is a t-reliable protocol from a to b with polynomial round and message complexity.*

5. CHARACTERIZING RELIABLE COMMUNICATION WITH ONE BYZANTINE PARTY

In this section, we consider the reliable transmission problem in the specific case of $t = 1$. A simple necessary condition for reliable transmission in this case is that the communication graph G_C is $(2, a, b)$-connected, and that $G = G_C \cup G_A$ is $(3, a, b)$-connected. We prove that in this case (i.e., $t = 1$) this condition is basically the characterization for reliable transmission.

LEMMA 5.1. *Let G_C be a $(2, a, b)$-connected communication graph. If G_C is connected and $G = G_C \cup G_A$ is $(3, a, b)$-connected, then for every $t_0, t_1 \in V \setminus \{a, b\}$ the pair $(\{t_0\}, \{t_1\})$ is not an (a,b) confusing pair.*

PROOF. Fix any $t_0, t_1 \in V \setminus \{a, b\}$. If there is a path in G_C that misses $\{t_0, t_1\}$, then by Property (1) of Definition 2.6 the pair $(\{t_0\}, \{t_1\})$ is not an (a, b) confusing pair. Otherwise, every path from a to b in G_C has a Byzantine vertex, t_0 or t_1, on it. Since G_C is $(2, a, b)$-connected, there are two disjoint paths from a to b in G_C, and there must be a Byzantine vertex on each of these paths. Hence, there is a path $P_{t_0,b}$ from t_0 to b that misses t_1 and there is a path $P_{t_1,b}$ from t_1 to b that misses t_0. Also, G_C is connected and for every $u \in V$ there is a path $P_{u,b}$ from u to b. If $P_{u,b}$ is not honest, then there is $i \in \{0, 1\}$ such that the prefix $\langle u, \ldots, t_i\rangle$ of $P_{u,b}$ misses $t_{\bar{i}}$, and $\langle u, \ldots, t_i\rangle, P_{t_i,b}$ is a semi-honest path from u to b. We conclude that for every $u \in V$ there is a semi-honest path $P_{u,b}$ from u to b.

Since G is 3 (a, b)-connected, there is a path P from a to b in G that misses $\{t_0, t_1\}$. We will prove that P is also a path in G^*, which implies by Definition 2.6 that $(\{t_0\}, \{t_1\})$ is not an (a, b) confusing pair. Assume towards contradiction that P is not in G^*. Hence, there is an authentication edge $e = \langle u, v\rangle$ and a honest path P', such that $\langle u, v\rangle, P'$ is a suffix of P, the edge e is not in E^*, and P' is in E^*.

We next check the conditions when $e \in E^*$ in Definition 2.4 of G^*. Since P misses $\{t_0, t_1\}$, then Property (1) holds. Since there is a semi-honest path $P_{u,b}$ from u to b, then the path $P_{u,b}, P'$ is a semi-honest path from u to v and therefore Property (2) holds. Finally, the path P' is a honest path from v to b and Property (3) holds. Hence, $e \in E^*$, contradiction. Thus, P is in G^* and $(\{t_0\}, \{t_1\})$ is not an (a, b) confusing pair in G. $\square$

THEOREM 5.2. *Let V' be the connected component of b in G_C, let E'_A be the set of authentication edges that connect vertices in V', and define $G' = \langle V', E_C \cup E'\rangle$. Then, 1-reliable communication from a to b is possible if and only if G_C is $(2, a, b)$-connected and G' is $(3, a, b)$-connected.*

PROOF. By lemma 5.1, the pair (T_0, T_1) is not an (a, b) confusing pair for all $T_0, T_1 \subseteq V \setminus \{a, b\}$ of size at most 1. By Theorem 2.7 it holds that (T_0, T_1)-reliable transmission from a to b is possible for all $T_0, T_1 \subseteq V \setminus \{a, b\}$ of size at most 1, which implies by [1] that there is a 1-reliable protocol from a to b. $\square$

Since the conditions of Theorem 5.2 are symmetric with respect to a and b we get that 1-reliable communication is symmetric.

COROLLARY 5.3. *1-reliable communication from a to b is possible if and only if 1-reliable communication from b to a is possible.*

6. RELIABLE COMMUNICATION IS NOT SYMMETRIC FOR $T \geq 2$

In the previous section we have seen a simple characterization for the case $t = 1$. In this section we show that the characterization for $t = 1$ can not be applied to $t \geq 2$. Moreover, we show that t-reliable communication is not symmetric.

LEMMA 6.1. *For every $t \geq 2$ there is a connected communication graph G_C and an authentication graph G_A such that G_C is $(t + 1, a, b)$-connected and $G = G_C \cup G_A$ is $(2t + 1, a, b)$-connected, however t-reliable communication from a to b is impossible.*

PROOF. For $t = 2$, consider $Graph1$ and the Byzantine sets described in Fig. 6. There is no semi-honest path from u to v, for every authentication edge $\langle u, v\rangle$. By Property (2) of the graph $Graph1^*$ this implies that $\langle u, v\rangle \notin E^*$ for every authentication edge $\langle u, v\rangle$ in $Graph1$. Since $Graph1^*$ is the communication

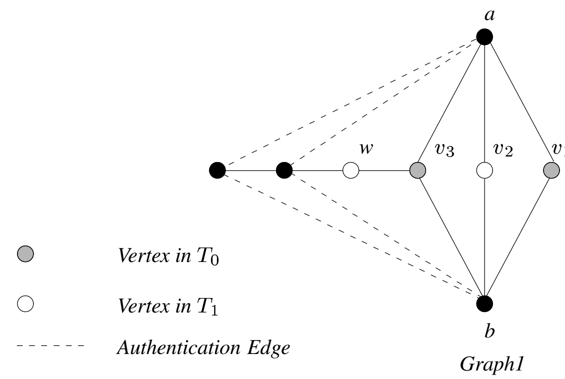

Figure 6: Confusing pairs for $t = 2$.

graph $Graph1$, and since there is no honest path from a to b in $Graph1^*$, by Definition 2.6 of a confusing pair, the pair (T_0, T_1) is an (a, b) confusing pair in $Graph1^*$, which implies that 2-reliable communication from a to b in $Graph1$ is impossible. For $t > 2$ consider the graph described in Fig. 7. In this graph the communication graph is $(2t - 1, b, a)$-connected and the union of the communication graph with the authentication graph is $(2t + 1, b, a)$-connected. Yet, we prove in Theorem 6.2 that t-reliable communication from b to a is impossible. $\square$

Beimel and Franklin [1] showed an example where fault restricted reliable communication is possible from a to b, but is impossible from b to a. However, in their example t-reliable communication is impossible in both directions. We present a stronger example in which t-reliable communication is possible from a to b, but impossible from b to a.

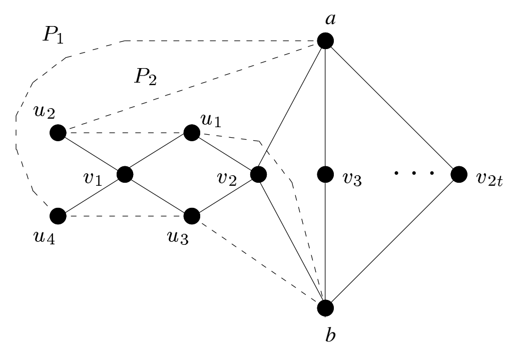

Figure 7: The graph G in which reliable communication is possible from a to b, but impossible from b to a.

THEOREM 6.2. *For every $t \geq 2$ there is a communication graph G_C and an authentication graph G_A such that t-reliable communication from a to b in $G = G_C \cup G_A$ is possible and reliable communication from b to a is impossible.*

PROOF. Consider the graph G described in Fig. 7 with $V = \{a, b, u_1, \ldots, u_4, v_1, \ldots, v_{2t}\}$. We first show that (T_0, T_1) is not an (a, b) confusing pair in G for all $T_0, T_1 \subseteq V \setminus \{a, b\}$. Fix any $T_0, T_1 \subseteq V \setminus \{a, b\}$ with size at most t. There are two cases:

1. There are Byzantine vertices on both P_1 and P_2: Consider the $2t - 1$ disjoint paths from a to b in the communication

graph of G^*. Since $|T_0 \cup T_1| \leq 2t$, there are at most $2t-2$ other Byzantine vertices in G, which implies that at least one of the $2t-1$ paths from a to b in the communication graph of G^* is clear from Byzantine vertices. Hence, there is a honest communication path in G^* from a to b.

2. There are no Byzantine vertices on either P_1 or P_2: Consider the path P_1. Regardless of whether v_1 or v_2 are Byzantine, there is a semi-honest path from u_4 to u_3, and there is a honest path $\langle u_3, b\rangle$ from u_3 to b. Therefore $\langle u_4, u_3\rangle \in E_2$. Finally, $\langle u_4, a\rangle \in E_3$ and all the edges on P_1 are added to G^*, which implies that P_1 is in G^*. We conclude that if there are no Byzantine vertices on P_1 then the path P_1 is in G^*. Symmetric arguments hold for P_2, and therefore either P_1 or P_2 is an honest path from a to b in G^*.

In both cases there is a honest communication path from a to b in G^* for all $T_0, T_1 \subseteq V \setminus \{a, b\}$ of size at most t, and we conclude that t-reliable communication from a to b is possible.

We now show that t-reliable communication from b to a is impossible. Fix $T_0 = \{v_1, v_{t+2}, \ldots, v_{2t}\}$ and $T_1 = \{v_2, \ldots, v_{t+1}\}$. We show that (T_0, T_1) is a confusing pair in G^* with respect to (b, a). Consider the path P_1, and note that $\langle a, u_4\rangle \notin E_1$ because there is no semi-honest path from a to u_4 in the communication graph of G. Furthermore, $\langle u_3, u_4\rangle$ is not added to E_1 since there is no honest path from either u_3 or u_4 to a. For the same reason $\langle u_3, b\rangle$ is not added to E_1. We conclude that no edge on P_1 is added to G^*. By symmetry, no edge on P_2 is added to G^*, and G^* is the communication graph of G. Since there is no honest path from b to a in G^*, this implies that (T_0, T_1) is a (b, a) confusing pair in G^*, which implies that t-reliable communication from b to a is impossible.

□

7. REFERENCES

[1] A. Beimel and M. Franklin. Reliable communication over partially authenticated networks. *Theoretical Computer Science*, 220:185–210, 1999.

[2] Y. Desmedt and Y. Wang. Secure communication in multi-cast channels: The answer to Franklin and Wright's question. *J. of Cryptology*, 14(2):121–135, 2001.

[3] Y. Desmedt and Y. Wang. Perfectly secure message transmission revisited. In L. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, Lecture Notes in Computer Science, pages 502–517. Springer-Verlag, 2002.

[4] D. Dolev. The Byzantine generals strike again. *J. of Algorithms*, 3:14–30, 1982.

[5] D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly secure message transmission. *J. of the ACM*, 40(1):17–47, 1993.

[6] C. Dwork, D. Peleg, N. Pippenger, and E. Upfal. Fault tolerance in networks of bounded degree. *SIAM J. on Computing*, 17(5):975–988, 1988.

[7] S. Even. *Graph Algorithms*. Computer Science press, 1979.

[8] M. J. Fischer, N. A. Lynch, and M. Merritt. Easy impossibility proofs for distributed consensus problems. *Distributed Computing*, 1(1):26–39, 1986.

[9] M. Franklin and R. N. Wright. Secure communication in minimal connectivity models. *J. of Cryptology*, 13(1):9–30, 2000.

[10] M. Franklin and M. Yung. Secure hyper-graphs: privacy from partial broadcast. In *Proc. of the 25th Annu. ACM Symp. on the Theory of Computing*, pages 36–44, 1993.

[11] O. Goldreich, S. Goldwasser, and N. Linial. Fault-tolerant computation in the full information model. In *Proc. of the 32nd Annu. IEEE Symp. on Foundations of Computer Science*, pages 447–457, 1991.

[12] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufman Publishers, 1997.

[13] T. Rabin and M. Ben-Or. Verifiable secret sharing and multi-party protocols with honest majority. In *Proc. of the 21st Annu. ACM Symp. on the Theory of Computing*, pages 73–85, 1989.

[14] H. M. Sayeed and H. Abu-Amara. Efficient perfectly secure message transmission in synchronous networks. *Information and Computation*, 126:53–61, 1996.

[15] G. J. Simmons. A survey of information authentication. In G. J. Simmons, editor, *Contemporary Cryptology, The Science of Information Integrity*, pages 441–497. IEEE Press, 1992.

[16] E. Upfal. Tolerating a linear number of faults in networks of bounded degree. *Information and Computation*, 115(2):312–320, 1994.