

On Exponential Lower Bound for Protocols for Reliable Communication in Networks

K. Srinathan¹ and C. Pandu Rangan² * and R. Kumaresan³ **

¹ Center for Security, Theory and Algorithmic Research
International Institute of Information Technology
Hyderabad India 500032
`srinathan@iiit.ac.in`

² Indian Institute of Technology
Chennai India
`rangan@cs.iitm.ernet.in`

³ Dept of Computer Science
University of Maryland
`ranjit@cs.umd.edu`

Abstract. This work deals with the problem of fault-tolerant communication over networks, some of whose nodes are corrupted by a centralized byzantine adversary. The extant literature's perspective of the problem of reliable communication, especially in networks whose topology is known, is that of a simple problem to which even some naive solutions (like message-flooding etc.) turn out to be reasonably efficient. In this paper, we give an example of a directed graph and a non threshold adversary structure, which will require every protocol for perfect reliable unicast to transmit exponential number of bits in order to *reliably* transmit a single bit.

Keywords: Perfect Reliable Unicast; Byzantine non threshold adversary; Efficiency.

1 Introduction

The problem of perfect reliable unicast is a fundamental problem in distributed networks ([5]). It deals with information transfer between a sender and a receiver in a network without any error. The network might have some faults, either in the form of link failures or node failures, which might interfere with the functioning of a protocol trying to achieve perfect reliable unicast. In this paper we consider only node failures. The failures are modeled by means of an adversary. We consider non-threshold adaptive adversaries for our analysis. By adaptive, we mean that the adversary fixes the set to be corrupted before the execution

* Work Supported by Project No. CSE/05-06/076/DITX/CPAN on Protocols for Secure Communication and Computation, Sponsored by Department of Information Technology, Govt. of India

** The work was done when the author was an under graduate student at IIT Madras

of the protocol but chooses the nodes that he would corrupt from the set as the protocol proceeds.

The extent to which these faults can affect the existence of a protocol for achieving reliable transfer is characterized in terms of graph structure in [3, 4]. In this work, we study how these faults can affect the efficiency of a protocol which tries to achieve reliable communication. We have given an example of a directed graph and an adversary structure for which all protocols for reliably transmitting a single bit need to transmit exponential number of bits. This shows that reliable communication is not always feasible even if it is possible.

2 Communication Model and Known Results

The network \mathcal{N} is modeled as a graph $\mathcal{G} = (V, E)$, where V denotes the node set (nodes are also known as players) and E refers to the links between the nodes. The faults in the network have been modeled as a centralized *Byzantine* adversary whose aim is to disrupt the functioning of any protocol for reliable communication to the maximum achievable level. Given below is the definition of the adversary structure ([4]), which explicitly denotes the set of sets that the adversary can corrupt.

DEFINITION *The adversary structure \mathcal{A}_{adv} is a monotone set of subsets of the player set V . The maximal basis of \mathcal{A}_{adv} denoted by \mathcal{A} is defined as the collection $\{A | A \in \mathcal{A}_{adv}, \exists X \in \mathcal{A}_{adv}, X \supset A\}$.*

It is assumed that all the nodes in the network know the network topology as well as the adversary structure. We further assume that the adversary chooses one of the sets from \mathcal{A}_{adv} for corrupting before the execution of the protocol but the protocol does not know the chosen set.

Reliable communication between the sender S and the receiver R is not always possible. The necessary and sufficient conditions for the existence of a protocol for reliable (secure) communication in this model is given by the following theorem([2]):

Theorem 1. *Perfectly secure message transmission from the sender \mathbf{S} to the receiver \mathbf{R} in the network \mathcal{N} is possible if and only if for any two sets $X, Y \in \mathcal{A}$, the deletion of the nodes in $X \cup Y$ from the network does not disconnect \mathbf{S} and \mathbf{R} .*

The possibility of the protocol is shown by a construction in [2]. The protocol given, is polynomial in the number of paths from S to R . In dense graphs, such a protocol is not feasible. However this does not rule out efficient protocols in such graphs. For example, consider a graph which is $(2t+1)-(S, R)$ -connected and \mathcal{A} consists of sets, all of which have cardinality t (threshold adversary). Irrespective of the number of paths in the graph, an efficient protocol exists([1]). One example of an efficient protocol would be to locate $2t+1$ disjoint paths by using a standard max-flow algorithm ([2]) and asking S to send the message along all the paths to R . Since there can be a maximum of t distorted versions of the message, the

receiver obtains the original message by taking majority among the messages received along the $2t + 1$ paths.

Another example graph is presented in Fig. 1. Consider the graph \mathcal{G}_1 in Fig. 1. We define the adversary structure $\mathcal{A} = \{\{x_1, x_2, \dots, x_k | x_i \in \{a_i, b_i\}\}, \{c\}\}$. Note that in the top layer, there are 2^k paths.

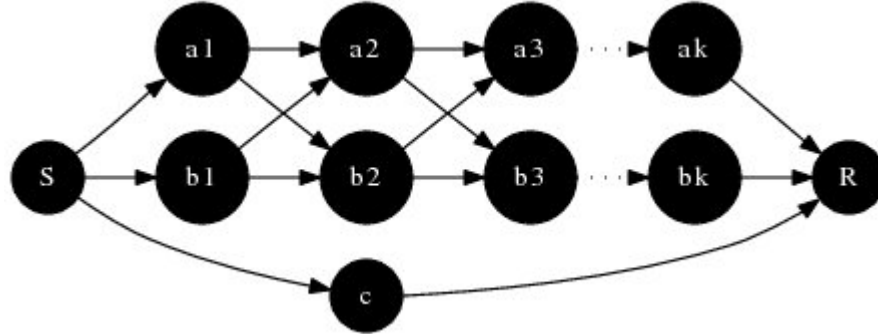


Fig. 1. An example graph where an efficient protocol for reliable communication exists.

However, there exists an efficient protocol in this case too, as given in Fig. 2.

Protocol for reliable unicast in \mathcal{G}_1

1. S sends the message m to a_1, b_1 and c .
2. c forwards m to R .
 a_1 and b_1 forward m to both a_2 and b_2 , if $k > 1$, else to R .
3. a_i, b_i forward m' if they receive m' from both their neighbors $\{a_{i-1}, b_{i-1}\}$ or $null$ if they receive different or no messages from their neighbors.
4. R receives three messages, say m_a from a_k , m_b from b_k and m_c from c . R recovers m as follows:

$$m = \begin{cases} m_a & m_a = m_b \text{ and } m_a \neq null \\ m_c & \text{otherwise} \end{cases}.$$

Fig. 2. Protocol for the graph given in Fig. 1.

Lemma 1. *The protocol given in Fig. 2 is correct.*

Proof. There are two cases to be considered.

Case 1. The adversary chooses to corrupt $\{x_1, x_2, \dots, x_k | x_i \in \{a_i, b_i\}\}$.

In this case, if there is no disruption at any stage, then $m_a = m_b = m_c$ and hence

R recovers m . Suppose the first disruption of the message occurs at some layer i , x_i changes the message from m to m' , then the honest player at layer $i + 1$ will get different messages and hence will forward *null*. Note that *null* will get forwarded by all the honest players through to R . On receiving *null*, R recovers m from m_c .

Case 2. The adversary chooses to corrupt c .

Since the top layer is not corrupted, $m = m_a = m_b$ and $m_a \neq \text{null}$ and R accepts m_a as the original message. \square

3 Reliable Unicast is not Always Efficient whenever Possible

Consider the directed graph \mathcal{G} shown in Fig. 3. The 6-tuple $\{a_i, b_i, c_i, d_i, e_i, f_i\}$ is referred to as *layer i* and the network \mathcal{H} (Fig. 4) consisting of only a_i 's and b_i 's is referred to as the *top band* (denoted by T) and the rest is referred to as the *bottom band* (denoted by B). Specifically the notation B_i is used to represent the set $\{c_i, d_i, e_i, f_i\}$. S is connected to R through each node in B_i . In T , for $1 \leq i < k$, there is a directed edge between a_i and a_{i+1}, b_{i+1} and the same holds for b_i . The graph consists of k layers. This makes a total of 2^k paths between S and R in T . The adversary structure for \mathcal{G} is introduced by presenting an instance of it. Define $\mathcal{B}_i = \{ \{a_i, c_i, d_i\}, \{b_i, c_i, e_i\}, \{a_i, e_i, f_i\}, \{b_i, d_i, f_i\} \}$ to be the adversary structure at layer i . Then the adversary structure for the graph \mathcal{G} is given as $\mathcal{A} = \{ A | A = \{A_1, A_2, \dots, A_k\}, A_i \in \mathcal{B}_i, 1 \leq i \leq k \}$.

Lemma 2. *There exist a protocol for reliable unicast between S and R in \mathcal{G} .*

Proof (PROOF). By Theorem 1, if the adversary could choose two sets X, Y such that $X, Y \in \mathcal{A}$ and removal of nodes belonging to $X \cup Y$ from \mathcal{G} disconnect S and R , then no protocol will exist. Let $X = \{X_1, X_2, \dots, X_k\}$ and $Y = \{Y_1, Y_2, \dots, Y_k\}$ with $X_i, Y_i \in \mathcal{B}_i$. To disconnect S and R , adversary has to remove all the nodes in B since otherwise there will be a direct path from S to R through one of them. Hence $X_i = \{a_i, c_i, d_i\}, Y_i = \{a_i, e_i, f_i\}$ or $X_i = \{b_i, c_i, e_i\}, Y_i = \{b_i, d_i, f_i\}$ or vice versa. In all these cases $\{a_i, b_i\} - (X_i \cup Y_i) \neq \phi$. Hence, there exists a path from S to R through T and adversary can never disconnect S and R . Hence, by Theorem 1, there exist a protocol. \square

Lemma 3. *For any adversarial strategy, R can identify in polynomial time and space either*

1. *message m transmitted by S , or*
2. *the set of corrupted nodes in T .*

Proof. Consider the following protocol for reliable unicast of m . Let S send m along B to R . Let m_x^i denote the message that R obtained from node $x_i, x_i \in B_i$. Let M_i be the 4-tuple $\{m_c^i, m_d^i, m_e^i, m_f^i\}$. If all the elements of M_i are equal for some i , then R receives m . Also, if any 3 of the 4 elements are equal, R can still

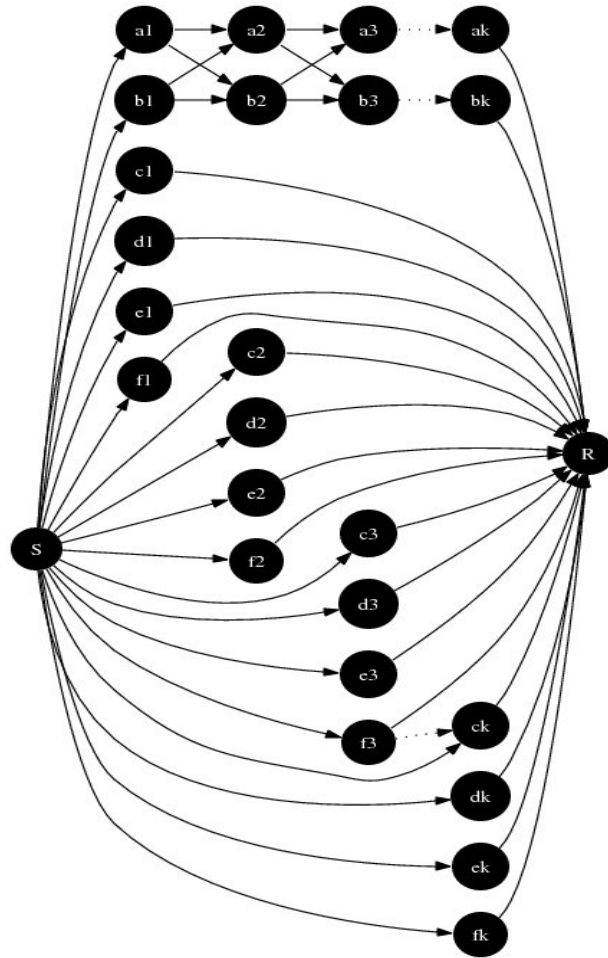


Fig. 3. The graph for which no efficient protocol exists given the adversary structure \mathcal{A} .

find out m , since there is no $A \in \mathcal{A}_{adv}$ such that $|A \cap \{c_i, d_i, e_i, f_i\}| > 2$, for $1 \leq i \leq k$. Hence corrupting less than 2 elements in each layer of the bottom band, leads to an unsuccessful adversarial strategy. The only strategy that would not allow R to know m would be to disrupt the message from both the nodes in each B_i . This would result in a 2-partition of B_i , say B_i^1 and B_i^2 . Now R can learn the set of corrupt nodes in each layer of T as $\mathcal{B}_i \cap \{\{a_i, b_i\} \times B_i^1\} \cap \{\{a_i, b_i\} \times B_i^2\}$. Hence R can learn either m or the set of corrupt nodes in T after receiving one message from each of polynomial number of paths from B in a single phase. \square

Lemma 4. *Atleast 1 bit must be transmitted through T for reliable unicast to be possible in \mathcal{G} .*

Proof. For the purpose of giving a proof by contradiction, assume that no bit is transmitted through T during a successful protocol for reliable unicast in \mathcal{G} . This would imply that all the messages pass only through B . Denote the subgraph induced by B on \mathcal{G} as \mathcal{G}_B . Consider $X = \{c_i, d_i\}, Y = \{e_i, f_i\}$. Clearly $X, Y \in \mathcal{A}_{adv}$ and removal of nodes belonging to $X \cup Y$ from \mathcal{G} disconnects S and R in \mathcal{G}_B . Hence by Theorem 1, there is no protocol for reliable unicast between S and R in \mathcal{G}_B . But by lemma 2, a protocol is possible in \mathcal{G} . Therefore, atleast 1 bit must be transmitted through T for reliable unicast to be possible in \mathcal{G} . \square

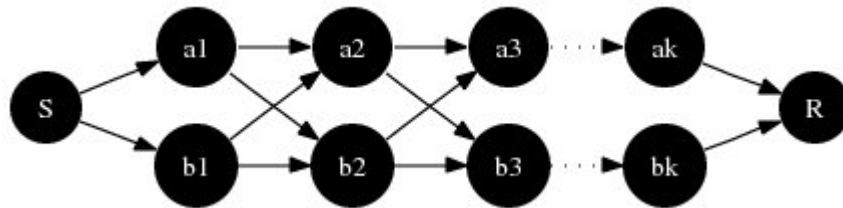


Fig. 4. Top layer of the graph given in Fig. 3. Every protocol for the reliable transmission of a bit require transmission of exponential number of bits through the network.

Lemma 5. *Given that only R knows the corrupted nodes in T , there exists no polynomial protocol for reliable unicast in \mathcal{H} .*

Proof. The proof proceeds by induction on k . The induction hypothesis is that “ R receives at least 2^{k-1} bits in every protocol for reliable unicast in \mathcal{H} ”. For purposes of clarity, m is assumed to be a single bit message. The base cases of $k = 1, 2$ have been discussed in detail.

$k = 1$: S simply sends m along both a_1 and b_1 . R knows the honest player and obtains m from him. R receives 1 bit and hence this base case is true.

$k = 2$: S sends m along both a_1 and b_1 . Assuming b_1 is a corrupt player, he sends to the honest player in the next layer (say b_2), whatever a_1 would send him if the message was \bar{m} . Player b_2 cannot distinguish between the case when a_1 is corrupt and the message is \bar{m} and the case when b_1 is corrupt and the message is m . This happens because b_2 does not have information on the corrupt nodes in T . Hence b_2 must send both the possibilities to R , who is able to distinguish between the two cases. Hence b_2 sends at least 2 bits and this base case is also true.

Assuming that the induction hypothesis is true for a graph \mathcal{H} with $1, 2, \dots, k-1$ layers, the proof that the induction hypothesis remains valid when the number of layers is k is as follows. First, note that in the worst case that the adversary chooses not to send any message to R via the corrupt player at layer k . Hence a perfectly reliable unicast protocol from S to R must essentially consist of a perfectly reliable unicast protocol from S to a_k , assuming b_k is the corrupt node at the k^{th} layer. Suppose b_{k-1} is the honest neighbor of a_k , then the strategy for a_{k-1} would be to transmit to a_k whatever b_{k-1} would transmit if the message was \bar{m} and not m . Since a_k cannot differentiate between the case when b_{k-1} is corrupt and m is the message and the case when a_{k-1} is corrupt and \bar{m} is the message, he has to send information on both the possibilities to R . By induction assumption, we have 2^{k-2} bits received by a_k in the transmission of m and another 2^{k-2} bits received in the transmission of \bar{m} . Hence a_k needs to send a total of 2^{k-1} bits to R . \square

Theorem 2. *Any protocol for reliable unicast in \mathcal{G} is infeasible.*

Proof. By lemma 3, there exists an adversarial strategy which allows R to know only the set of corrupt nodes in T without knowing anything about m . This implies that the precondition for lemma 4 is satisfied, and hence, along with lemma 5, it can be easily seen that there exists no polynomial protocol for reliable unicast in \mathcal{G} . \square

4 Efficient Protocols in Undirected Graphs

The graph \mathcal{G} in the previous section is directed. This leads us to the question whether there exists undirected graphs where every protocol for reliable unicast is of exponential space or time complexity. The notation \mathcal{G}_u and \mathcal{H}_u is used to denote the undirected versions of \mathcal{G} and \mathcal{H} . In general the subscript u is used to denote the corresponding undirected version.

Undirected graphs usually help in providing greater interaction. It is known that interaction, by itself cannot help in making *possible* a protocol for reliable unicast exist in a given network and an adversarial structure. But protocols using interaction (i.e. protocols which are not single phase), can be more efficient than those which do without it ([6]). For example, consider the graph \mathcal{G}_1 given in Fig. 5.

Let the honest player at i^{th} layer be denoted by $h_i, h_i \in \{a_i, b_i\}$. The power of interaction comes when in \mathcal{G}_u , R is able to tell h_k , who h_{k-1} is, and hence

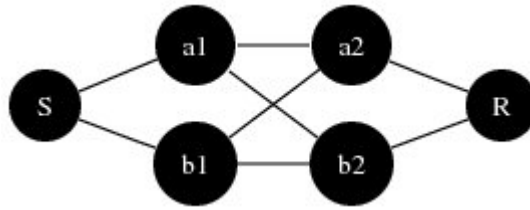


Fig. 5. Undirected graph with 2 layers.

avoid doubling of messages at h_k . h_k simply forwards the message obtained from h_{k-1} to R . Presented in Fig. 6 is a protocol in \mathcal{G}_1 which makes use of 2-way communication. It assumes that R knows the honest nodes h_i at each layer.

Protocol for reliable unicast in \mathcal{G}_1

1. S sends the message m to a_1, b_1 .
 2. R sends id of h_1 to h_2 .
 3. h_1 forwards m to a_2, b_2 .
 4. h_2 forwards whatever it receives from h_1 to R and discards whatever was received from $\{a_1, b_1\} - \{h_1\}$.
-

Fig. 6. A reliable unicast protocol for the graph given in Fig. 5.

In the above protocol, we have outlined the roles of all the honest nodes. The proof that the protocol works is also trivial. Also note that there is no possibility of doubling of messages at any stage in the protocol. Note that lemmas 2-4 hold for \mathcal{G}_u . In addition to these, we have the following lemma too.

Lemma 6. *If R knows the set of corrupted nodes in T , then S can learn this information.*

Proof. The above lemma is similar to lemma 3. Note that the graph is symmetric with respect to S and R . Replacing m by m' (which denotes the set of corrupted nodes in T), S by R and vice versa, will complete the proof. \square

We hope that like its directed counterpart, \mathcal{G}_u too will require every protocol for PRU to transmit exponential number of bits.

Conjecture 1. In the graph \mathcal{G}_u , every protocol for reliable unicast will need to transmit atleast 2^{k-1} bits to transmit a single bit reliably between S and R .

Note that the conjecture takes into account both the forward and backward transmissions within a link.

4.1 Another Example Graph

We now give another example of a graph (Fig. 7) for which no efficient protocol might exist. The adversary structure for \mathcal{G} is introduced by presenting an instance of it.

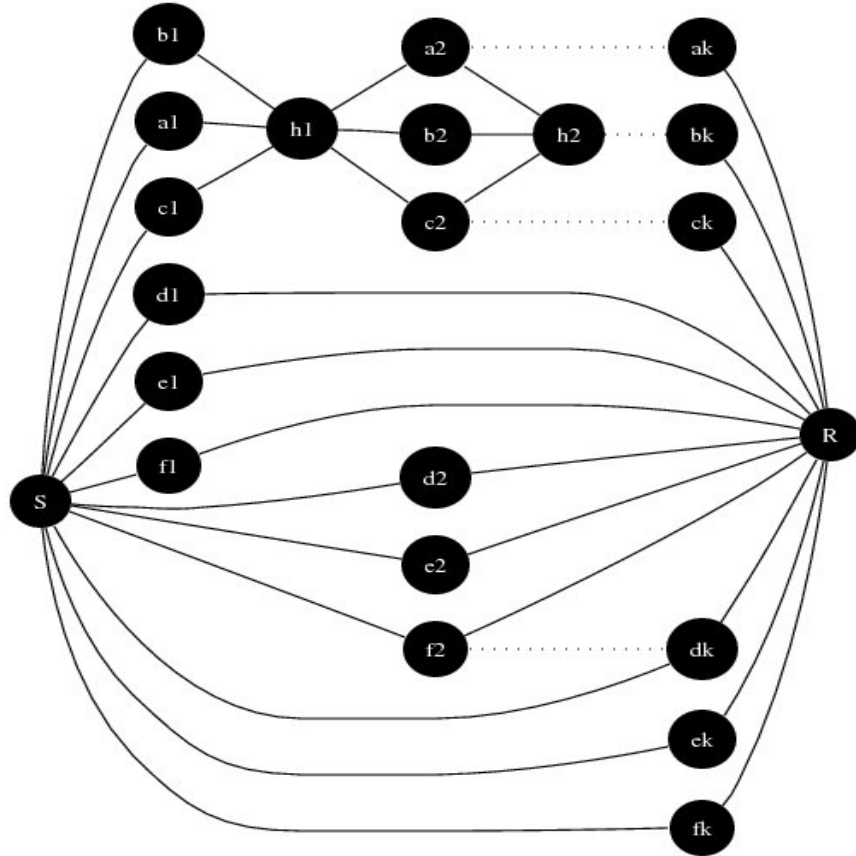


Fig. 7. Another graph in which no efficient PRU protocol may exist.

Define

$$\mathcal{B}_i = \{ \{a_i, b_i, d_i\}, \{a_i, c_i, f_i\}, \{a_i, d_i, e_i\}, \\ \{b_i, c_i, e_i\}, \{b_i, e_i, f_i\}, \{c_i, d_i, f_i\} \}$$

to be the adversary structure at layer i . Then the adversary structure for the graph \mathcal{G} is given as $\mathcal{A} = \{ A | A = \{A_1, A_2, \dots, A_k\}, A_i \in \mathcal{B}_i, 1 \leq i \leq k \}$.

5 Conclusions and Open Problems

Efficiency of Reliable Communication is an important problem as it finds every-day use widespread across the internet in the form of e-mails. Hence, there is a need for characterizing graphs which have efficient protocols and for an algorithm that determines whether the input graph has an efficient protocol or not. In most of the cases, just eliminating distorted messages, with the help of fault knowledge may lead to a efficient protocol. Note that any adversary structure with polynomial number of possibilities or a graph with polynomial number of paths has an efficient protocol (by brute force) if the adversary structure satisfies Theorem 1.

We have shown an exponential lower bound for the communication complexity of the reliable communication problem in a directed network. A natural question that arises is whether there exists an undirected graph, in which all protocols for reliable unicast are infeasible.

Other closely related problems are finding the necessary and sufficient conditions for the existence of a polynomial protocol in graphs whose network topology is only partially known. Solutions under different models have been posed to this problem in [1, 7] but the general problem still remains open.

References

1. Mike Burmester and Yvo Desmedt. Secure communication in an unknown network using certificates. In *ASIACRYPT*, pages 274–287, 1999.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms Second Edition*. The MIT Press and McGraw-Hill Book Company, 2001.
3. Danny Dolev, Cynthia Dwork, Orli Waarts, and Moti Yung. Perfectly secure message transmission. *J. ACM*, 40(1):17–47, 1993.
4. M V N Ashwin Kumar, Pranava R. Goundan, K Srinathan, and C. Pandu Rangan. On perfectly secure communication over arbitrary networks. In *PODC '02: Proceedings of the twenty-first annual symposium on Principles of distributed computing*, pages 193–202, New York, NY, USA, 2002. ACM Press.
5. N. Lynch. Distributed algorithms. *Morgan Kaufmann*, 1996.
6. Hasan Md. Sayeed and Hosame Abu-Amara. Efficient perfectly secure message transmission in synchronous networks. *Inf. Comput.*, 126(1):53–61, 1996.
7. Lakshminarayanan Subramanian, Randy H. Katz, Volker Roth, Scott Shenker, and Ion Stoica. Reliable broadcast in unknown fixed-identity networks. In *PODC '05: Proceedings of the twenty-fourth annual ACM SIGACT-SIGOPS symposium on Principles of distributed computing*, pages 342–351, New York, NY, USA, 2005. ACM Press.