How Far Must You See to Hear Reliably

[Extended Abstract]

Pranav K Vasishta[*]
vasishta@research.iiit.ac.in

Anuj Gupta[*]
anujgupta@research.iiit.ac.in

Prasant Gopal[†]
prasant@csail.mit.edu

Piyush Bansal[*]
piyush_bansal@research.iiit.ac.in

Rishabh Mukherjee[*]
rishabh_m@research.iiit.ac.in

Poornima M[*]
mpoornima@research.iiit.ac.in

Kannan Srinathan[*]
srinathan@iiit.ac.in

Kishore Kothapalli[*]
kkishore@iiit.ac.in

Abstract

We consider the problem of probabilistic reliable communication (PRC) over synchronous networks modeled as directed graphs in the presence of a Byzantine adversary when players' knowledge of the network topology is not complete. We show that possibility of PRC is extremely sensitive to the changes in players' knowledge of the topology. This is in complete contrast with earlier known results on the possibility of perfectly reliable communication over undirected graphs where the case of each player knowing only its neighbours gives the same result as the case where players have complete knowledge of the network. Specifically, in either case, $(2t+1)$-vertex connectivity is necessary and sufficient, where t is the number of nodes that can be corrupted by the adversary [DDWY93, SKR$^+$05]. We introduce a novel model for quantifying players' knowledge of network topology, denoted by $\mathcal{TK}$. Given a directed graph G, influenced by a Byzantine adversary that can corrupt up to any t players, we give a necessary and sufficient condition for possibility of PRC over G for any arbitrary topology knowledge $\mathcal{TK}$. It follows from our main characterization theorem that knowledge of up to $d = \lfloor \frac{n-2t}{3} \rfloor + 1$ levels is sufficient for the solvability of honest player to honest player communication over any network over which PRC is possible when each player has complete knowledge of the topology. We also show the existence of networks where PRC is possible when players have complete topology knowledge but it is not possible when the players do not have knowledge of up to $d = \lfloor \frac{n-2t}{3} \rfloor + 1$ levels.

Keywords: reliable communication, topology knowledge, synchronous networks, directed graphs, Byzantine adversary

[*] Center for Security, Theory and Algorithmic Research(CSTAR), International Institute of Information Technology, Hyderabad, 500032, India.

[†] Computer Science and Artificial Intelligence Laboratory(CSAIL), Massachusetts Institute of Technology, Cambridge, MA 02139, USA.

1 Introduction

Reliable Communication is one of the fundamental problems and one among the keenly studied problems of distributed computing. The problem of reliably communicating between a Sender **S** and a receiver **R** over a network $\mathcal{N}$ in the presence of an adversary that can corrupt up to t players (denoted by a t-adversary) and make them behave arbitrarily is of great relevance in almost all practical circumstances. In general, while studying the problem of reliable communication, one assumes knowledge of complete topology for each player. We consider the problem of communicating reliably (with negligible error) between **S** and **R** in the presence of a t-adversary over synchronous directed networks when players possess only partial knowledge of the topology.

Reliable transmission of messages between two honest (not corrupted by t-adversary) players is of utmost importance because several distributed computing problems will be rendered unsolvable in the absence of reliable transmission between honest players. We focus on probabilistic reliable communication (PRC) between two honest players and give the necessary and sufficient conditions for the possibility of PRC over a synchronous directed network for any arbitrary topology knowledge. By PRC, we mean, PRC in synchronous networks (modelled as a directed graph) in the presence of a Byzantine adversary. In the problem of PRC over a synchronous network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ where $\mathbb{P}$ is the set of vertices and $\mathcal{E}$ denotes the set of arcs/edges in the network, the sender **S** $\in \mathbb{P}$ wishes to send a message m to the receiver $\mathbf{R} \in \mathbb{P}$ in a robust manner such that the message is correctly received by **R** with a very high probability, in spite of the presence of up to t Byzantine-faulty nodes in $\mathcal{N}$.

Generally knowledge of the topology is mentioned as players having knowledge of their neighbors, or their neighbors' neighbors, that is up to a level d. We consider knowledge of up to level 1 as knowledge of both in-neighbours and out-neighbours of the corresponding players.

In this paper, we refer to the notion of knowledge possessed by a node about the graph G by Topology Knowledge $\mathcal{TK}$ at that node, which we define as a collection of possible graphs with one of them being the actual graph. One can easily note that the notion of $\mathcal{TK}$ is relevant only in the presence of a suitable adversary. In the absence of a suitable adversary, each node can let the rest of the players (of course only as far as the connectivity permits!) know its view of the network in order to end up with a global common picture of the topology for each of the connected components respectively. Consequently, while the distributed *complexity* may increase, the distributed *computability* is unaffected. However, in the presence of the adversary, any amount of communication would entail only an (adversary controlled) approximation of the actual topology, thereby perhaps affecting even the distributed computability. Intuitively though, the adversary can at best completely hide the edges between two faulty players and if lucky, succeed in partially hiding even the edges with one faulty end-node. However, useful messages are seldom transmitted via the aforementioned edges. This gives a feeling that distributed *computability* may not be affected — in fact, for the case of perfectly reliable communication it has been proved that the knowledge of one's neighbors alone is as sufficient as the knowledge of the global topology; specifically, $(2t+1)$-connectivity is necessary and sufficient irrespective of $\mathcal{TK}$, where up to t players are Byzantine faulty (t-adversary)[SKR$^+$05]. Counter-intuitively, for the case of probabilistic reliable communication, we show that the optimal fault-tolerance heavily *depends* on $\mathcal{TK}$.

In [VGG$^+$09], Pranav *et al.* show that knowledge of partial topology knowledge affects the possibility of PRC by giving a specific example. In this paper, we give the complete characterization for the possibility of PRC over any given network modeled as directed graphs under the influence of a (static) t-Byzantine adversary when the players have arbitrary topology knowledge $\mathcal{TK}$. We use our definition $\mathcal{TK}$ which is set, to arrive at the characterization. To appreciate the consequences of our results, we also arrive at a distance measure, that **R** must see up to $d = \lfloor \frac{n-2t}{3} \rfloor + 1$ levels

in order to ensure the possibility of PRC protocol over those networks for which PRC is possible given complete topology knowledge. That is to say, PRC between any two honest players over any network is possible if the receiver amongst the players' sees distance $d = \lfloor \frac{n-2t}{3} \rfloor + 1$ levels provided PRC were solvable in the case where players have complete knowledge of the topology.
Related Work: The problem of perfectly reliable communication tolerating a t-adversary over undirected graphs is introduced by Dolev *et al.* [DDWY93]. PRC over undirected graphs was introduced by Franklin and Wright [FW98, FW00]. Desmedt and Wang [DW02] were the first to study the problem of reliable communication over directed networks. $(2t+1)$-connectivity between **S** and **R** is necessary and sufficient for all the problems given in the prequel. Shankar *et al.* [SGSR08] show that the results for the case of possibility of PRC in directed graphs are markedly different from the earlier results, which appear consistent. This difference is because their results show that connectivity requirements on the graph for the possibility of PRC over directed graphs as being the receiver **R**-specific (See Necessity proof of Theorem 4.1). However, they study PRC for the case where every player has complete knowledge of the topology. In this sequence of study, we study the problem of PRC in directed graphs for partial knowledge of the topology. The line of study that we take originated in the paper by Srinathan and Pandurangan in [SR06], where the authors show the conditions for the possibility of PRC over directed graphs for a non-threshold adversary.

Contributions: Our Contributions through this paper are manifold: (1) We completely characterize the problem of PRC between **S** and **R** over arbitrary synchronous directed networks in the presence of a t-adversary, when each player has partial knowledge of the topology. (See Theorem 4.1) (2) It follows from our main characterization theorem that knowledge of up to $d = \lfloor \frac{n-2t}{3} \rfloor + 1$ levels is sufficient for the solvability of honest player to honest player communication over any network over which PRC is possible when each player has complete knowledge of the topology. (See Corollary 4.1.1) (3) We also show the existence of networks where PRC is possible when players have complete topology knowledge but it is not possible when the players do not have knowledge of up to $d = \lfloor \frac{n-2t}{3} \rfloor + 1$ levels. (See Corollary 4.1.2) (4) We give a strong definition of topology knowledge from which it is possible to extract information as to whether the players' have a knowledge of up to k levels. (See Definition 1). (5) Arriving at our main characterization require us to take a detour in building tools which enable us to handle arbitrary topology knowledge. These tools make extensive utilization of our Definition for Topolology Knowledge (See Definition 2). The tools in themselves are for distributed computing problems with certain properties (See Table 1) and PRC satisfies the relevant conditions. We give them as Bridge Theorems, Theorem 3.1 and Theorem B.1.

Our focus in this paper is on finding the possibility of PRC protocols, and we do not focus on the complexity of the protocols. Our constructions to prove the possibility of PRC protocols lead us to protocols with super-polynomial complexity. We remark that finding polynomial-time solutions for the problem is quite challenging and we do not rule out the possibility of an exponential lower bound.

2 Model and Definitions

The network is modeled as a directed graph $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ where $\mathbb{P}$ is the set of vertices and $\mathcal{E}$ denotes the set of arcs/edges in the directed graph. We model the nodes/players (as well as the adversary) as interactive Turing machines with unbounded computing power. The system is assumed to be synchronous, that is, the protocol is executed in a sequence of *rounds* wherein in each round, a player can perform some local computation, send new messages to his out-neighbors, receive the messages sent in that round by his in-neighbors (and if necessary perform some more local computation), in that order. In the graph, we assume that the channels are *secure*. In other words, if $(u, v) \in \mathcal{E}$

then *the player u can securely send a message to player v in one round.* During the execution, the adversary may corrupt up to any t players. We work with a (static) Byzantine adversary that may completely control up to t players (denoted by t-adversary) and make them behave in arbitrary fashion. Every honest player that receives a message from its in-neighbor knows the sender as it can identify the channel along which the message is received. All messages to be transmitted are chosen from a finite field $\mathbb{F}$.

2.1 Definitions

Definition 1 (Topology Knowledge of *player i* ($\mathcal{TK}_i$)) *In a given network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$, we define* Topology Knowledge of player i $\mathcal{TK}_i$ *as the knowledge possessed by* player i *about the network $\mathcal{N}$. It is represented by a set of graphs,i.e. $\mathcal{TK}_i = \{G^i_{k_i}\}$, with a condition that one of the graphs from the set $\mathcal{TK}_i$ is the actual graph $\mathcal{N}$ where $1 \leq k_i < 2^{|\mathbb{P}|^2}$, that is, $1 \leq |\mathcal{TK}_i| < 2^{|\mathbb{P}|^2}$.*

Notice that when you define topology knowledge of a player this way, the player is in possession of adjacency matrices of each of the graphs, of which one of them is actual graph. So, if for any player j (j could be i also), the entries of rows (out-edges) and columns (in-edges) are the same across all the matrices, then we say that *player i* has knowledge of up to level 1 with respect to the players j. If for every player j, the entries for rows and columns are the same across all adjacency matrices, then it is as good as knowing the complete topology.

Definition 2 (Topology Knowledge $\mathcal{TK}$) *We define the notion Topology Knowledge $\mathcal{TK}$ as the collection of all the individual $\mathcal{TK}_i$'s (for all the players in $\mathbb{P}$). Specifically, $\mathcal{TK} = \{\mathcal{TK}_i | i \in \mathbb{P}\}$.*

Definition 3 (k-sized $\mathcal{TK}_i$ and k-sized $\mathcal{TK}$) *A k-sized $\mathcal{TK}_i$ is defined as $\mathcal{TK}_i$ where $|\mathcal{TK}_i| = k$. If there exists an integer k such that every $\mathcal{TK}_i$ in the collection $\mathcal{TK}$ is ℓ-sized for some $\ell \leq k$, then the topology knowledge $\mathcal{TK}$ is defined as a k-sized $\mathcal{TK}$.*

At the outset, we assume each *player i* knows the following: (1) Set of vertices in $\mathcal{N}, i.e, \mathbb{P}$ (2) Topology Knowledge of *player i*, $\mathcal{TK}_i$. We also assume the worst-case scenario where the adversary knows all the $\mathcal{TK}_i$'s for $i \in \mathbb{P}$ as well as is aware of the actual graph G.

Note 1 *When each of the graphs is written in $G = (V, E)$ form, we note that $G^i_k = (\mathbb{P}, \mathcal{E}^i_{k_i})$, and only the edge-set keeps changing across the graphs for each of the players. Therefore, we can replace every graph $G^i_{k_i}$ in the set $\mathcal{TK}_i$ with the set of its edge-sets $\{\mathcal{E}^i_{k_i}\}$ for each* player i *to make matters more convenient. We will be using $\mathcal{TK}_i$ synonymously with $\{\mathcal{E}^i_{k_i}\}$ from now on.*

3 Bridge Theorems

As per the Definition 2, $\mathcal{TK}$ can be a set containing varying topology knowledge edge sets for each player, thereby indicating that the extent of knowledge possessed by each player is different. This is how arbitrary it can get. Working with this arbitrariness could bewilder more than clarify what can be done with. But the use of set theoretic notation for $\mathcal{TK}$ has its own advantages which can be seen from the two theorems in this section. These theorems act as the bridge theorems for our main characterization theorem, Theorem 4.1.

In this section, we give a theorem in which we show that those problems π that satisfy certain conditions have a property that, over a network $\mathcal{N}$ working with an N-sized $\mathcal{TK}$ is equivalent to working with (several) 2-sized $\mathcal{TK}$ with respect to the solvability of π. This equivalence works only for those problems that satisfy the conditions in the Table 1 (See Appendix A).These conditions are a prerequisite for the correctness of majority voting which we employ to show the equivalence. Majority voting is applied to the outputs obtained on solving π for various different-sized $\mathcal{TK}$s.

We begin by giving followed by a definition which we use in our proof for the first bridge theorem in our attempt to solve for the possibility of PRC in the setting that we work.

3.1 Bridge Theorem 1: N-sized $\mathcal{TK}$ to 2-sized $\mathcal{TK}$

We begin with the following definition.

Definition 4 *Given a graph $G = (\mathbb{P}, E)$ with N-sized $\mathcal{TK} = \{\mathcal{TK}_{P_1}, \mathcal{TK}_{P_2}, \ldots, \mathcal{TK}_{P_{|\mathbb{P}|}}\}$, $N > 2$, where each $\mathcal{TK}_{P_i} = \{\mathcal{E}_1^{P_i}, \mathcal{E}_2^{P_i}, \ldots, \mathcal{E}_k^{P_i}\}$, $k \leq N$, and a 2-sized topology knowledge $\mathcal{Z} = \{\mathcal{Z}_{P_1}, \mathcal{Z}_{P_2}, \ldots, \mathcal{Z}_{P_{|\mathbb{P}|}}\}$ with each $\mathcal{Z}_{P_i} = \{A_0^{P_i}, A_1^{P_i}\}$, $1 \leq i \leq |\mathbb{P}|$, such that one of the $A_j^{P_i}$'s is exactly the edge-set E (the edge-set of G) while the other is an edge-set (different from E) that is present in $\mathcal{TK}_{P_i}$. In other words, there exists a $j \in \{0, 1\}$ such that $E = A_j^{P_i}$ and $A_{\bar{j}}^{P_i} \in \mathcal{TK}_{P_i}$ for every player P_i, we say that $\mathcal{Z}$ is derived from $\mathcal{TK}$.*

Note that an N-sized $\mathcal{TK}$ can have up to $(N-1)^{|\mathbb{P}|}$ distinct 2-sized topology knowledge sets derived from it.

Theorem 3.1 *A protocol Π that solves a problem π, satisfying either IO Condition and Input Honesty Condition or IO Condition and Output Agreement Condition in Table 1, over a network $\mathcal{N}$ with N-sized $\mathcal{TK}$ $(N > 2)$ influenced by an adversary $\mathcal{A}$ exists if and only if, for each of the 2-sized topology knowledge $\mathcal{Z}_j$ $(1 \leq j \leq (N-1)^{|\mathbb{P}|})$, that can be derived from $\mathcal{TK}$, there exists a protocol solving π with topology knowledge $\mathcal{Z}_j$.*

Proof: Necessity: We need to prove that if π cannot be solved with one of the 2-sized topology knowledge sets $\mathcal{Z}_j$ derived from $\mathcal{TK}$ then π cannot be solved with $\mathcal{TK}$. The proof follows from the Definition 3. Notice that, all valid 2-sized topology knowledge sets derived from $\mathcal{TK}$ are included, by definition, in an N-sized topology knowledge set. That is, Z_j is also an N-sized topology knowledge set. If a problem π cannot be solved with Z_j derived from $\mathcal{TK}$ in the presence of an adversary $\mathcal{A}$, then the strategy of the adversary is to change the topology knowledge from $\mathcal{TK}$ to Z_j by suitably communicating to the nodes the information needed to eliminate the edge-sets at which topology knowledge of each player in $\mathcal{TK}$ differs from the corresponding topology knowledge in Z_j. This naturally means that π cannot be solved with $\mathcal{TK}$.

Sufficiency: We need to prove the statement: If there exists a protocol Π_j that can solve the problem π for the j^{th} 2-sized topology knowledge $\mathcal{Z}_j$ $(1 \leq j \leq (N-1)^{|\mathbb{P}|})$ derived from $\mathcal{TK}$, then there exists a protocol Π that can solve π with topology knowledge $\mathcal{TK}$. We give a proof by induction on the size of $\mathcal{TK}_{P_i}$ of player P_i $(1 \leq i \leq |\mathbb{P}|)$.

Base Case: In this case, $N = 3$. That means $\mathcal{TK}_{P_i}$ is a 3-sized set. Let $\mathcal{TK}_{P_i} = \{\mathcal{E}_1^{P_i}, \mathcal{E}_2^{P_i}, \mathcal{E}_3^{P_i}\}$, $1 \leq i \leq |\mathbb{P}|$. We can partition $\mathcal{TK}_{P_i}$ into 3 sets, each of which is of size less than 3 such that every element in $\mathcal{TK}_{P_i}$ occurs in two of the three sets. Since any one of the three elements of $\mathcal{TK}_{P_i}$ could be the actual edge-set, two of the following three sets will be the 2-sized topology knowledge sets derived from $\mathcal{TK}_{P_i}$: $\mathcal{Z}_A^{P_i} = \{\mathcal{E}_1^{P_i}, \mathcal{E}_2^{P_i}\}$, $\mathcal{Z}_B^{P_i} = \{\mathcal{E}_1^{P_i}, \mathcal{E}_3^{P_i}\}$, $\mathcal{Z}_C^{P_i} = \{\mathcal{E}_2^{P_i}, \mathcal{E}_3^{P_i}\}$.

A j^{th} 2-sized topology knowledge $\mathcal{Z}_j$ $(1 \leq j \leq (N-1)^{|\mathbb{P}|})$ is a collection of the 2-sized topology knowledge sets of all players P_i. In $\mathcal{Z}_j$, for player P_i, the 2-sized topology knowledge is one of $\mathcal{Z}_A^{P_i}$, $\mathcal{Z}_B^{P_i}$, $\mathcal{Z}_C^{P_i}$ and is denoted by $\mathcal{Z}_{\alpha_i}^{P_{ij}}$, where $\alpha_i \in \{A, B, C\}$. Therefore, $\mathcal{Z}_j = \{\mathcal{Z}_{\alpha_1}^{P_{1j}}, \mathcal{Z}_{\alpha_2}^{P_{2j}}, \ldots \mathcal{Z}_{\alpha_{|\mathbb{P}|}}^{P_{|\mathbb{P}|j}}\}$.

We assume that there exists a protocol Π_j that can solve the problem π for the j^{th} 2-sized topology knowledge $\mathcal{Z}_j$. This Π_j can be considered to be a collection of protocols of each individual player P_i. For each player P_i, let there be protocols $\Pi_A^{P_i}$ defined over $\mathcal{Z}_A^{P_i}$, $\Pi_B^{P_i}$ defined over $\mathcal{Z}_B^{P_i}$, $\Pi_C^{P_i}$ defined over $\mathcal{Z}_C^{P_i}$, such that each of the protocols is defined under the assumption that the 2-sized topology knowledge set over which it is defined is a valid 2-sized topology knowledge set (those sets in which one of the edge-set is the actual edge-set), and the collection of all valid protocols for each player defined over valid 2-sized topology knowledge set solves the problem π. Any collection of valid protocols defined over a valid 2-sized topology knowledge set (those sets in which one of the edge-set is the actual edge-set) of each individual player forms a protocol like Π_j. Corresponding to the respective 2-sized topology knowledge for each player P_i in $\mathcal{Z}_j$, one could write $\Pi_j = \{\Pi_{\alpha_1}^{P_{1j}}, \Pi_{\alpha_2}^{P_{2j}}, \dots \Pi_{\alpha_{|\mathbb{P}|}}^{P_{|\mathbb{P}|j}}\}$ for $\alpha_i \in \{A, B, C\}$. The subscript j is used to indicate that the j^{th} protocol Π_j is running over the j^{th} valid 2-size topology knowledge $\mathcal{Z}_j$.

Let $(\alpha_{i\beta}, \alpha_{i\gamma}) \in \{(A, B), (B, C), (C, A)\}$ indicate the pair of valid 2-sized topology knowledge sets $\mathcal{Z}_{\alpha_{i\beta}}^{P_i}$,$\mathcal{Z}_{\alpha_{i\gamma}}^{P_i}$ for a player P_i. For these two sets, a protocol for P_i is part of the collection of protocols Π_β and Π_γ over two 2-sized topology knowledge sets $\mathcal{Z}_\beta$ and $\mathcal{Z}_\gamma$. Therefore, protocols like $\Pi_A^{P_i}$, $\Pi_B^{P_i}$, $\Pi_C^{P_i}$ exist.

Recall that the problem π satisfies **IO Condition** and **Input Honesty Condition** or **IO Condition** and **Output Agreement Condition** in Table 1. IO Condition and Input Honesty Condition together imply that there is a unique output for a given input for the problem π. IO Condition and Output Agreement Condition force the inputs to be consistent so as to make the outputs same.

Notice that using the three protocols, $\Pi_A^{P_i}$, $\Pi_B^{P_i}$ and $\Pi_C^{P_i}$, problem π can be solved even if the player P_i is not aware of the actual topology. Let there be three protocols Π_β, Π_γ, Π_δ where in each of them, the protocol for P_i is $\Pi_{\alpha_{i\beta}}^{P_i}$, $\Pi_{\alpha_{i\gamma}}^{P_i}$ and $\Pi_\kappa^{P_i}$ where $\kappa \in \{\{A, B, C\} \setminus \{\alpha_{i\beta}, \alpha_{i\gamma}\}\}$ respectively. Specifically, all the players run all the three protocols Π_β, Π_γ and Π_δ and obtain three outputs O_1, O_2 and O_3 respectively. Note that the actual graph is part of all the individual players' topology knowledge in at least two of the three cases. That is, actual graph is the input to at least two of the three protocols. Thus, at least two of the three outputs must be same and equal to the output that a protocol solving π would produce. Thus, in a similar manner, every player can take the majority of the three outputs and thereby solve π. Thus, a protocol for π with topology knowledge $\mathcal{TK}$ exists if and only if a protocol for solving π exists with each of the two valid topology knowledge sets (i.e. those that contain the actual edge-set) among $\mathcal{Z}_A^{P_i}$, $\mathcal{Z}_B^{P_i}$, $\mathcal{Z}_C^{P_i}$. Thus, we can say that if some protocol solves a problem for each of its 2-sized $\mathcal{TK}$s derived from 3-sized $\mathcal{TK}$, then Π solves the problem with 3-sized $\mathcal{TK}$.

Induction Hypothesis: Let us suppose that the statement is true for up to any m-sized $\mathcal{TK}$. That is to say, If π is solvable for each of its 2-sized $\mathcal{TK}$s derived from the m-sized $\mathcal{TK}$, then π can be solved with topology knowledge $\mathcal{TK}$.

Induction: Let $\mathcal{TK}_i$ be $m + 1$-sized $\mathcal{TK}_i$. Since $m + 1 > 3$, we can partition $\mathcal{TK}_i$ of every player P_i into 3 sets, say X, Y, Z each of which is of size less than $m + 1$ such that every element in $\mathcal{TK}_i$ occurs in two among X, Y and Z. Therefore two of X, Y and Z make a valid m-sized $\mathcal{TK}$. From our induction hypothesis, we know that π can be solved with an m-sized topology knowledge M if it can be solved for each of the 2-sized topology knowledge sets derived from M. In the proof for the Base Case condition, we replace 2-sized topology knowledge with m-sized topology knowledge set, and the set $\{A, B, C\}$ with $\{X, Y, Z\}$ and $(\alpha_{i\beta}, \alpha_{i\gamma}) \in \{(A, B), (B, C), (C, A)\}$ with $(\alpha_{i\beta}, \alpha_{i\gamma}) \in \{(X, Y), (Y, Z), (Z, X)\}$. We proceed along the same lines as in the proof for Base Case. This way, we can show how a protocol that solves π for each of the m-sized $\mathcal{TK}$ derived from $m + 1$-sized $\mathcal{TK}$ can solve π for $m + 1$-sized $\mathcal{TK}$. Thus, the statement is true for an $m + 1$-sized $\mathcal{TK}$.Therefore, by induction, it is true $\forall m \in \mathbb{N}$. ■

3.2 Arbitrary to Uniform $\mathcal{TK}$

From the prequel, it is enough to characterise the possibility of PRC for a 2-sized $\mathcal{TK}$. For a set of players $\mathbb{P}$, a 2-sized $\mathcal{TK} = \{\mathcal{TK}_1, \mathcal{TK}_2, \ldots, \mathcal{TK}_{|\mathbb{P}|}\}$, where each $\mathcal{TK}_i = \{\mathcal{E}_0^i, \mathcal{E}_1^i\}$, $\forall i \in \mathbb{P}$. Notice that each player can have a different $\mathcal{TK}_i$, and working with varying $\mathcal{TK}_i$s can become cumbersome. In our characterization, we avoid working directly with 2-sized $\mathcal{TK}$ for this reason. In this section, we show that for any problem π, working with a 2-sized $\mathcal{TK}$ can be brought down to working with a modified $\mathcal{TK}$, say, $\mathcal{TK}'$ which is formed by the intersection of each of the $\mathcal{TK}_i$s in 2-sized $\mathcal{TK}$ and a set denoted by 2-sized $\mathcal{TK}_G$ which in all respects has the properties of a 2-sized $\mathcal{TK}_i$ and is known globally to all the players. There are $2^{|\mathbb{P}|^2}$ possibilities for 2-sized $\mathcal{TK}_G$. We show that solving π using a 2-sized $\mathcal{TK}$ is possible if and only if π can be solved for all the $2^{|\mathbb{P}|^2}$ possibilities for $\mathcal{TK}'$. Following this, we give a necessary and sufficient condition for the possibility of PRC given $\mathcal{TK}'$. Owing to space constraints, we are moving this to Appendix B

Given that we have these Bridge Theorems: Theorem 3.1 and Theorem B.1 and since the problem of PRC satisfies the conditions under which the theorems are true, the task of solving PRC can now be taken up on the 2-sized uniform $\mathcal{TK}$. Following this, we construct back the conditions for PRC for the case of an N-sized $\mathcal{TK}$ by following the path taken to reduction in the reverse direction.

4 Main Characterization Theorem for PRC

We begin with a few definitions and lemmas. Our proofs use rigorous path analysis of paths between the sender **S** and the receiver **R**.

Definition 5 (Strong Path) *A sequence of vertices $v_1, v_2, v_3, \ldots, v_k$ is said to be a strong path from v_1 to v_k in the network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ if for each $1 \leq i < k$, $(v_i, v_{i+1}) \in \mathcal{E}$. Furthermore, we assume that there vacuously exists a strong path from a node to itself.*

Definition 6 (t-(S,R,)-strong-connectivity) *A digraph is said to be t-(**S**,**R**)-strong-connected if the graph is such that there exists at least t vertex disjoint strong paths from **S** to **R**.*

Definition 7 (Semi-Strong Path) *A sequence of vertices $v_1, v_2, v_3, \ldots, v_k$ is said to be a semi-strong path from v_1 to v_k in the network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ if there exists a $1 \leq j \leq k$ such that the sequence v_j to v_1 as well as v_j to v_k both are strong paths in the network. We call the vertex v_j as the head of the semi-strong path.*

Definition 8 (Weak Path) *A sequence of vertices $v_1, v_2, v_3, \ldots, v_k$ is said to be a weak path from v_1 to v_k in the network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ if for each $1 \leq i < k$, either $(v_i, v_{i+1}) \in \mathcal{E}$ or $(v_{i+1}, v_i) \in \mathcal{E}$. Furthermore, we assume that there vacuously exists a weak path from a node to itself.*

Definition 9 (PRC Protocol) *Let $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ be a network, with topology knowledge $\mathcal{TK}$, under the influence of a Byzantine adversary that may corrupt up to any t players. We say that a protocol for transmitting a message from **S** to **R** is (t, δ)-reliable if for any valid adversary strategy, the probability that **R** outputs **m** given that **S** has sent **m**, is at least δ where the probability is over the random inputs of all the players and random inputs of the adversary.*

Whenever we refer to a PRC Protocol in the rest of the paper, it is assumed to be (t, δ)-reliable.

Definition 11 (Critical Combination) *Given the following: A network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$, where $\mathbb{P}$ is the set of players, and $\mathcal{E}$, the set of edges between the players in $\mathbb{P}$. Two players identified as* **S** *(denoting sender) and* **R** *(denoting receiver) such that $\{\mathbf{S}, \mathbf{R}\} \in \mathbb{P}$. Two t-sized sets - B_1 and B_2, such that $B_1 \subset \mathbb{P}$, $B_2 \subset \mathbb{P}$ in the network $\mathcal{N}$. A 2-sized $\mathcal{TK}_G = \{E_0, E_1\}$ for the network $\mathcal{N}$. A 2-sized $\mathcal{TK}$ for the network $\mathcal{N}$. A set X of players, $X = \{i | \mathcal{TK}_i' = \mathcal{N}$ and $i \in (\mathbb{P} \setminus B_1 \ \cup B_2 \cup \mathbf{R}$-group$)\}$; Let $C = (\mathbb{P} \setminus B_1 \cup B_2 \cup \mathbf{R}$-group); X is a set of players in C which know the actual graph $\mathcal{N}$ after knowing $\mathcal{TK}_G$. $X \ \cap \ \mathbf{R}$-group $= \emptyset$. A set $Z \subset C$ of players that are part of all weak paths from* **S** *to* **R** *and those players that have a semi-strong path from itself to* **R**.
Network $\mathcal{N}$ is said to be in a Critical Combination if any of the following hold:

- *Either B_1 or B_2 cut across all strong paths between* **S** *and* **R***s.*
- *$B_1 \cup B_2$ cut across all weak paths between* **S** *and* **R**.
- *$\exists W$ such that every weak path p that avoids both B_1 and B_2 between* **S** *and* **R** *has a node, say w, that has both its adjacent edges (along p) directed inwards and $w \in W$ and the following hold:*
  - *Both B_1 and B_2 are vertex cut-sets between w and* **R**. *In other words, every strong path from w to* **R** *passes through both B_1 and B_2.*
  - *For $\alpha \in \{0, 1\}$, B_1 is a vertex cut-set between all nodes in ($W \cup (Z \cap X)$) and* **R** *in the edge-set $E_\alpha \in \mathcal{TK}_G$ and B_2 is a vertex cut-set between all nodes in ($W \cup (Z \cap X)$) and* **R** *in the edge-set $E_{\overline{\alpha}} \in \mathcal{TK}_G$.*

Definition 10 ((Player p)-Group) *(Player p)-group is defined with respect to two t-sized subsets of $\mathbb{P}$, say B_1 and B_2. A p-group in graph G is the set of all players q (including p) such that q has a strong path from it to p not passing through any node in $(B_1 \cup B_2)$. It also contains any player w that can reliably communicate even though the path from it to* **R** *passes through B_1 or B_2.*

Note 2 *Following Definition 11, network $\mathcal{N}$ can be divided into four components for a given two t-sized sets, B_1 and B_2 as follows:* **R***-group, B_1, B_2 and the rest of the players together as one, say $C = \mathbb{P} \setminus (B_1 \cup B_2 \cup$* **R***-group). Since the sender* **S** *and the receiver* **R** *are honest, we consider the case when $\mathbf{S} \in C$. The only other possibility is $\mathbf{S} \in$* **R***-group, in which case the protocol for PRC is obvious.* **R***-group has no knowledge of the actual graph $\mathcal{N}$, and each player in* **R***-group has the same topology knowledge as that of the globally declared 2-sized $\mathcal{TK}_G$, made of two edge-sets $\{E_0, E_1\}$. Set $X \subset C$. Set $Z \subset C$. Players in $\mathbb{P} \setminus (X \cup$* **R***-group) all have the same topology knowledge $\{E_0, E_1\}$.*

Theorem 4.1 (Main Theorem) *For $\delta > \frac{1}{2}$, a (t,δ)-reliable PRC protocol between* **S** *and* **R** *in the network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$ with a 2-sized $\mathcal{TK}$ tolerating a t-adversary exists if and only if for every B_1, B_2 of size at most t, such that $B_1 \in \mathbb{P}$, $B_2 \in \mathbb{P}$ in the network, Critical Combination [Definition 11] does not occur in $\mathcal{N}$.*

Proof. Necessity: For Necessity,we must show that a network $\mathcal{N}$ that is in Critical Combination guarantees that no PRC protocol from **S** to **R** exists in $\mathcal{N}$. We now take each of the conditions described for a network to be in Critical Combination (Definition 11) and show that no PRC protocol can exist between S and R when the condition is true.

Note that if the $\mathcal{TK}_i'$ of all the players, following the inputs given in the Definition 11, is a singleton set, that is all the players have identified the actual graph $\mathcal{N}$ of the network, then the requirement for the PRC protocol is, as per the conditions in [SGSR08]. The results in [SGSR08] have a strong *receiver* **R**-specificity. With our results we gain a better insight into this **R**-specificity. In fact, we show that the knowledge of the actual graph or the absence of it for the receiver **R** is the key for the existence or non-existence of the PRC protocol. The following four lemmas shall capture the requirement of our necessity proof.

Lemma 4.1.1 *Following Definition 11, if either B_1 or B_2 cut across all strong paths between* **S** *and* **R** *in $\mathcal{N}$, then a PRC protocol between* **S** *and* **R** *does not exist.*

Proof: It is obvious to see that in this case, an adversary that can corrupt up to any t players can corrupt the set B_1 or B_2 that cuts across all strong paths between **S** and **R**, and thereby disconnect the two in which case no PRC protocol can ever exist. ■

Lemma 4.1.2 *Following Definition 11,if $B_1 \cup B_2$ cut across all weak paths between* **S** *and* **R***, then a (t,δ)-reliable PRC protocol ($\delta > \frac{1}{2}$) between* **S** *and* **R** *does not exist.*

Lemma 4.1.3 *Following Definition 11, if $\exists W$ such that every weak path p that avoids both B_1 and B_2 between* **S** *and* **R** *has a node, say w, that has both its adjacent edges (along p) directed inwards and $w \in W$ and both B_1 and B_2 are vertex cut-sets between w and* **R***, then a PRC protocol between* **S** *and* **R** *does not exist. In other words,if every strong path from w to* **R** *passes through both B_1 and B_2, then a (t,δ)-reliable PRC protocol ($\delta > \frac{1}{2}$) between* **S** *and* **R** *does not exist.*

Proofs of Lemmas 4.1.2 and 4.1.3 are given in Appendix B.

Lemma 4.1.4 *Following Definition 11, if $\exists W$ such that every weak path p that avoids both B_1 and B_2 between* **S** *and* **R** *has a node, say w, that has both its adjacent edges (along p) directed inwards and $w \in W$ and for $\alpha \in \{0, 1\}$, B_1 is a vertex cut-set between all nodes in ($W \cup (Z \cap X)$) and* **R** *in the edge-set $E_\alpha \in \mathcal{TK}_G$ and B_2 is a vertex cut-set between all nodes in ($W \cup (Z \cap X)$) and* **R** *in the edge-set $E_{\overline{\alpha}} \in \mathcal{TK}_G$. then a ($t$,$\delta$)-reliable PRC protocol ($\delta > \frac{1}{2}$) between* **S** *and* **R** *does not exist.*

Proof Outline: The proof of this lemma gives the topology knowledge effect on the possibility of (t,δ)-reliable PRC protocol for a given network modeled as a directed graph. The proof is by contradiction.

Assume that a protocol Π solves PRC (with error probability less than $\frac{1}{2}$) from **S** to **R** in the edge-set E_0 tolerating t-adversary. If one considers Π as a collection of programs to be run by each of the players in the graph G, then define another protocol Π' such that in Π', the players **S**, **R**, and players in **R**-group run the same programs as they run in the protocol Π, the players in sets B_1 and B_2 swap the programs that they run in Π, and the players in the set $(W \cup (Z \cap X))$ runs the program in Π with one change: wherever it is to send its message to B_1 in Π, it sends its message to B_2 in Π'. This protocol Π' is clearly a protocol to solve PRC from **S** to **R** in the edge-set E_1 tolerating t-adversary.

Consider two executions F_1 of Π in the edge-set E_0 and F_2 of Π' in edge-set E_1. In both executions the vertices in **R**-group hold the random inputs $\{\rho_u | u \in$ **R**-group $\}$. In the execution $\mathbf{F}_\alpha \in \{F_1, F_2\}$, the Byzantine set B_α is corrupt and the message m_α is transmitted by **S**, the random inputs of the vertices in $(C \cup B_{\overline{\alpha}})$[1] are $\{\rho_u | u \in (C \cup B_{\overline{\alpha}})\}$. The behavior of the Byzantine set B_α in the execution F_α is to send no message whatsoever to $C \cup B_{\overline{\alpha}}$ and to send to **R**-group exactly the same messages that are sent to **R**-group by the honest B_α in the execution $F_{\overline{\alpha}}$. In order for the Byzantine set B_α to behave as specified in the execution F_α, the adversary needs to simulate the

[1] We denote $\overline{1} = 2$ and vice-versa;

behavior of $(C \cup B_\alpha)$ in the execution $\mathbf{E}_{\overline{\alpha}}$. To achieve this task,the adversary simulates round-by-round the behavior of the vertices in $(C \cup B_\alpha)$ for the execution $F_{\overline{\alpha}}$ using $\{\rho_u | u \in (C \cup B_\alpha)\}$ as the random inputs for the vertices in $(C \cup B_\alpha)$. At the beginning of each round, each simulated player has a history of messages that it got in the simulation of the previous rounds and its simulated local random input. The simulated player sends during the simulation the same messages that the honest player would send in the original protocol in the same state. The simulated messages that (players in) B_α sends to **R** are really sent by the players. All other messages are used only to update the history for the next round. The messages which are added to the history of each simulated vertex are the real messages that are sent by players in **R**-group and the simulated messages that are sent by the vertices in $(C \cup B_\alpha)$. No messages from $B_{\overline{\alpha}}$ are added to history. The history of messages of each simulated vertex in execution F_α is the same as the history of the vertex in execution $F_{\overline{\alpha}}$. Therefore, the messages sent by B_1 and B_2 to members of **R**-group in both executions are exactly the same and the members of **R**-group and in particular the receiver **R** receive and send the same messages in both executions. Thus, the receiver **R** cannot distinguish whether the set B_1 is corrupt and the message transmitted by **S** is m_1 or the set B_2 is corrupt and the message transmitted by **S** is m_2. Now, consider all the pairs of executions where the random inputs range over all possible values. In each pair of executions, whenever **R** accepts the correct message in one execution it commits an error in the other. Thus, for any strategy by **R** for choosing whether to receive m_1 or m_2 there is some α such that when m_α is transmitted, the receiver accepts m_α with probability at most $\frac{1}{2}$. ■

This completes the necessity part of the proof for Theorem 4.1.

Proof. Sufficiency: For Sufficiency, Network $\mathcal{N}$ that is not in Critical Combination guarantees the existence of a PRC protocol from S to R in $\mathcal{N}$. So, we prove by giving a protocol and its proof of correctness. Owing to space constraints, we move the discussion of the protocol to Appendix C. ■

4.1 Corollaries

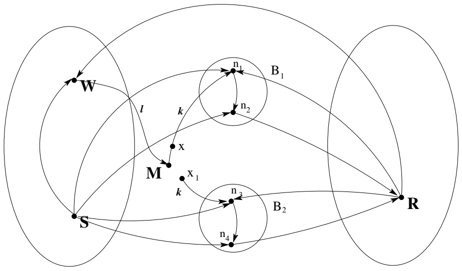

Figure 1: Network $\mathcal{N}$

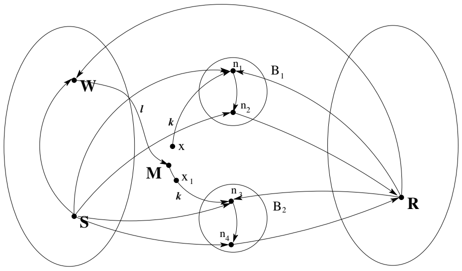

Figure 2: Network $\mathcal{N}'$

Corollary 4.1.1 *There exist networks over which, a PRC protocol between a Sender* **S** *and the Receiver* **R** *in the network tolerating a t-adversary exists, if every player in the network has complete knowledge of the topology but does not exist if in the network, each player does not have knowledge of up to d_{min}-levels (hops) (both for in-edges as well as out-edges) where $\mathbf{d_{min}} = \lfloor \frac{n-2t}{3} \rfloor + 1$, where n is the total number of players in the network.*

Proof: We give a proof by construction. The Figure 3 represents the state of a network $\mathcal{N}$ similar to what is given in Note 2. The network $\mathcal{N}$ is constructed such that it satisfies the conditions for the possibility of PRC tolerating a t-adversary when each player has complete knowledge of the topology. Let there be n nodes in the network. The number of nodes in the path between W and M is l. Notice that the sets B_1 and B_2 cut across all strong paths between **S** and **R**. There are k nodes in the path connecting M to a node in B_1 and k nodes in a path from M to B_2 which is disconnected at node X_1. Sets B_1 and B_2 are sets of t nodes each, of which the adversary corrupts on of them. Both B_1 and B_2 are such that they contain nodes of the kind n_1, n_3 that form part of disjoint weak paths from **S** to **R**. They also contains nodes of the kind n_2, n_4 that form part of disjoint strong paths from **S** to **R**, totalling $t+1$. There is a weak path between **S** and **R** passing through W. Notice that, in the network, **R** knows W, n_1, n_2, n_3, n_4 with knowledge of up to a single hop. **S** knows W,n_1,n_2,n_3,n_4 with a knowledge of up to a single hop. That is there are no nodes in between the paths from **S** and **R** to these nodes. Clearly, for this network, PRC protocol exists in the presence of a complete topology knowledge, because the underlying undirected graph is $(2t+1)$-S,R connected and it has $(t+1)$-strong paths for any number of nodes l in the path between W and M.

Let $l >= k-1$ for the network. Clearly, for this network if we count the number of nodes and equate it to n, we get: $n = 4 + 2k + l + 2t$, which, when you take for the minimum value of l, $n = 2t + 3k + 3$.

In such a network $\mathcal{N}$, notice that the adversary can thwart the possibility of PRC in the absence of knowledge of $\mathbf{d_{min}}$ levels, because, for knowledge of up to $\mathbf{d_{min}} - 1$ hops, it can ensure indistinguishable views for the receiver over two executions, E_a and E_b by constructing another network similar to that of $\mathcal{N}$ - $\mathcal{N}'$ (see Figure 4). The adversarial strategy is exactly as described in the proof of Lemma 4.1.4, and it is as good as saying that $\mathcal{TK}_R$ contains both $\mathcal{N}$ and $\mathcal{N}'$. In the construction, it is only when each player has knowledge of up to $\mathbf{d_{min}}$ levels that **R** distinguishes between $\mathcal{N}$ and $\mathcal{N}'$. ■

Corollary 4.1.2 *For any network, existence of a PRC protocol between* **S** *and* **R** *in the network tolerating a t-adversary when each player has complete knowledge of the topology implies existence of a PRC protocol between* **S** *and* **R** *in the network tolerating a t-adversary when each player has knowledge of up to $\mathbf{d_{min}}$-levels (hops) (both for in-edges as well as out-edges) where $\mathbf{d_{min}} = \lfloor \frac{n-2t}{3} \rfloor + 1$, where n is the total number of players in the network.*

Proof: Proof for the Corollary 4.1.2 is given in Appendix E.

5 Conclusion

We have provided a complete characterization for the problem of Probabilistic Reliable Communication given topology knowledge as a parameter. The significance of our results can be understood from the following implications: (a) *Generalization:* Our results are a strict generalization of the existing results for probabilistic reliable communication [SGSR08]; (b) *The "Randomization-effect"*: It is well known that for perfectly reliable communication, fault-tolerance is independent of nodes' knowledge of the network topology [SKR$^+$05]; we show that in the case of probabilistic reliable communication, fault-tolerance is extremely sensitive to changes in the knowledge of network topology. c) *Optimization*: Our results may be used to answer the question: *what is the optimal fault-tolerance that is achievable in reliable communication for a specified $\mathcal{TK}$ and vice-versa.*

References

[DDWY93] D. Dolev, C. Dwork, O. Waarts, and M. Yung. Perfectly Secure Message Transmission. *Journal of the Association for Computing Machinery (JACM)*, 40(1):17–47, January 1993.

[DW02] Y. Desmedt and Y. Wang. Perfectly Secure Message Transmission Revisited. In *Proceedings of Advances in Cryptology EUROCRYPT '02*, volume 2332 of *Lecture Notes in Computer Science (LNCS)*, pages 502–517. Springer-Verlag, 2002.

[FMS03] Paola Flocchini, Bernard Mans, and Nicola Santoro. Sense of direction in distributed computing. *Theor. Comput. Sci.*, 291(1), 2003.

[FRS99] Paola Flocchini, Alessandro Roncato, and Nicola Santoro. Backward consistency and sense of direction in advanced distributed systems. In *PODC '99: Proceedings of the eighteenth annual ACM symposium on Principles of distributed computing*. ACM, 1999.

[FW98] M. Franklin and R. N. Wright. Secure Communication in Minimal Connectivity Models. In *Proceedings of Advances in Cryptology EUROCRYPT '98*, volume 1403 of *Lecture Notes in Computer Science (LNCS)*, pages 346–360. Springer-Verlag, 1998.

[FW00] M. Franklin and R.N. Wright. Secure Communication in Minimal Connectivity Models. *Journal of Cryptology*, 13(1):9–30, 2000.

[KGSR02] M.V.N.A. Kumar, P. R. Goundan, K. Srinathan, and C. Pandu Rangan. On perfectly secure communication over arbitrary networks. In *Proceedings of the 21st Symposium on Principles of Distributed Computing (PODC)*, pages 193–202, Monterey, California, USA, July 2002. ACM Press.

[SGSR08] Bhavani Shankar, Prasant Gopal, Kannan Srinathan, and C. Pandu Rangan. Unconditionally reliable message transmission in directed networks. In *SODA '08: Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1048–1055, 2008.

[SKR$^+$05] 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 symposium on Principles of distributed computing*, New York, NY, USA, 2005. ACM.

[SR06] Kannan Srinathan and C. Pandu Rangan. Possibility and complexity of probabilistic reliable communications in directed networks. In *Proceedings of 25th ACM Symposium on Principles of Distributed Computing (PODC'06)*, 2006.

[VGG$^+$09] Pranav K. Vasishta, Prasant Gopal, Anuj Gupta, Piyush Bansal, and Kannan Srinathan. Brief announcement:Topology knowledge affects probabilistic reliable communication. In *Proceedings of 28th ACM Symposium on Principles of Distributed Computing (PODC'09)*, 2009.

A Size Reduction Conditions

The following are the conditions that the problems must satisfy for the equivalence of N-sized $\mathcal{TK}$ to 2-sized $\mathcal{TK}$ to work:

The conditions and the corresponding equivalence between solvability of problems with N-sized $\mathcal{TK}$ and the solvability of problems using some k-sized $\mathcal{TK}$. where $k <= N$, are as follows:

- **IO Condition:** The problem should be such that its Input-Output relation must be a function, that is, for a given input, there is only a unique output.
- **Input Honesty Condition:** Is there a requirement for the input givers to be honest?
  - If yes, then there is an equivalence that can be shown from N-sized $\mathcal{TK}$ to 2-sized $\mathcal{TK}$.
  - If not,**Output Agreement Condition:** Is there a requirement for agreement amongst the outputs?
    - If yes, **Output Secrecy Condition:**Is there a requirement for the secrecy of outputs?
      - If yes, then there is an equivalence that can be shown from N-sized $\mathcal{TK}$ to k-sized $\mathcal{TK}$, where $k > 2$. (This is our conjecture)
      - If not, then there is an equivalence that can be shown from N-sized $\mathcal{TK}$ to 2-sized $\mathcal{TK}$.
    - If not,then the equivalence is dependent on the Input-Output relation. In other words, it is sensitive to input-output relation and equivalence is not generically obtained.

Table 1: Size Reduction Conditions

A.1 Reductions Conditions satisfied by PRC

For the problem of Probabilistic Reliable Communication, the input-output relation is a function. The function is to output a yes/no depending on whether reliable message transmission takes place between a sender **S** and a receiver **R** or not, where the inputs are a graph that models the network, and the adversary that operates in the network. Since we have not yet given the condition in the presence of partial topology knowledge, let every player in the network know the complete topology. Shankar *et al.* in [SGSR08] have given the conditions that are required for the occurrence of PRC between **S** and **R**, which is the function that operates on this problem. The function gives an answer yes/no which is unique for the given input. So, PRC satisfies the IO Condition in the Reduction Conditions mentioned in Table 1.

One may consider input giver, here **S** to be honest, or dishonest in PRC. If the input giver is considered to be honest, then the Input Honesty Condition is satisfied by PRC.

Even if the input giver is dishonest, since the PRC protocol has only one output from the receiver, the Output Agreement condition is satisfied. Rather, the IO Condition enforces Output Agreement condition in the absence of Input Honest condition. For PRC, Output Secrecy is not a requirement. So, it need not satisfy the secrecy requirement.

Thus PRC falls under both the categories that satisfy: (1) IO Condition and Input Honesty (2) IO Condition and Output Agreement.

B Bridge Theorem 2 : From 2-sized $\mathcal{TK}$ to $\mathcal{TK}'$

We begin with the following definitions before we give the Bridge Theorem 2:

Definition 12 (Global $\mathcal{TK}_G$ and k-sized $\mathcal{TK}_G$) *For a given network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$, a global $\mathcal{TK}_G$ is defined as a globally published topology information accessible to all the players in the network $\mathcal{N}$. Its representation and properties are similar to that of $\mathcal{TK}_i$. It is represented as a set of graphs, i.e. $\mathcal{TK}_G = \{G_i\}$ with the condition that one of the graphs in $\mathcal{TK}_G$ is the actual graph $\mathcal{N}$, where $1 \leq |\mathcal{TK}_G| < 2^{|\mathbb{P}|^2}$. If $|\mathcal{TK}_G| = k$, then the $\mathcal{TK}_G$ shall be called k-sized $\mathcal{TK}_G$. Similar to Note 1, $\mathcal{TK}_G$ can be represented using the corresponding edge-sets of the graphs in it.*

Definition 13 ($\mathcal{TK}'_i$ and $\mathcal{TK}'$) *Given a network $\mathcal{N} = (\mathbb{P}, \mathcal{E})$, a $\mathcal{TK}_G$ and a $\mathcal{TK}$, then we define $\mathcal{TK}'_i$ as the updated topology knowledge of* player i *from knowing $\mathcal{TK}_G$. Collection of all such $\mathcal{TK}'_i$s forms $\mathcal{TK}'$,i.e., $\mathcal{TK}' = \{\mathcal{TK}_i \cap \mathcal{TK}_G\}$ $\forall i \in \mathbb{P}$. Similar to Note 1, $\mathcal{TK}'$ can be represented using the corresponding edge-sets of the graphs in it.*

Theorem B.1 *A protocol Π that solves a problem π over a network $\mathcal{N}$ with 2-sized $\mathcal{TK}$ influenced by an adversary $\mathcal{A}$ exists if and only if, for each of the $\mathcal{TK}'$ sets formed from the $2^{|\mathbb{P}|^2}$ possibilities for the 2-sized $\mathcal{TK}_G$, there exists a protocol solving π with $\mathcal{TK}'$.*

Proof: Only-If part: This is the easier part. It is evident that the topology knowledge $\mathcal{TK}'$ is higher than the topology knowledge $\mathcal{TK}$, since every $\mathcal{TK}_i$ is a superset of $\mathcal{TK}'_i = \mathcal{TK}_i \cap \mathcal{TK}_G$. Therefore, if a protocol Π works correctly over $\mathcal{TK}$, it would vacuously be a correct protocol over $\mathcal{TK}'$ too.
If part: Let there exist protocols for problem π for each of the valid 2-sized $\mathcal{TK}_G$s. We now show that this implies that there exists protocols for π for any valid 3-sized $\mathcal{TK}_G$. Specifically, let a 3-sized $\mathcal{TK}_G$ be $\mathcal{T} = \{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3\}$. Consider the three 2-sized subsets, namely, $X_1 = \{\mathcal{E}_1, \mathcal{E}_2\}$, $X_2 = \{\mathcal{E}_2, \mathcal{E}_3\}$ and $X_3 = \{\mathcal{E}_1, \mathcal{E}_3\}$. Note that at least two of the above three X_i's are valid global $\mathcal{TK}_G$'s (that is they contain the actual edge set). Suppose the players execute the protocol for solving π for each of these two $\mathcal{TK}_G$s, their outputs must exactly match since the problem is solved over the same network with the same inputs! However, the players are unaware of which of the two X_i's are valid global $\mathcal{TK}_G$s. It turns out that this does not matter since the players could execute protocols for all the three X_i's and perform a majority voting on the outputs to obtain the correct output for problem π for the 3-sized $\mathcal{TK}_G$, namely $\mathcal{T}$.

By induction, one can now solve the problem π for any given 4-sized $\mathcal{TK}_G$ (since any 4-sized $\mathcal{TK}_G$ can be split into three subsets of size ≤ 3 such that at least two of them valid and yield exactly the same output). Continuing further, we find that solving π with any m-sized $\mathcal{TK}_G$ (where $2 < m \leq 2^{|\mathbb{P}|^2}$) is possible if and only if π is solvable for each of its 2-sized subset $\mathcal{TK}_G$s. Notice that the case of $m = 2^{|\mathbb{P}|^2}$ is nothing but the case where $\mathcal{TK}$ is exactly equal to $\mathcal{TK}'$. Hence the theorem. ■

C Proof of Lemma 4.1.2, Lemma 4.1.3, Lemma 4.1.4

Lemma C.0.1 *Following Definition 11,if $B_1 \cup B_2$ cut across all weak paths between* **S** *and* **R**, *then a PRC protocol between* **S** *and* **R** *does not exist.*

Note 2 gives us the glimpse of the state of the network $\mathcal{N}$ following Definition 11.

It is evident from the definition of **R**-group that there do not exist vertices $u \in C$ and $v \in$ **R**-group, such that the edge (u, v) is in $\mathcal{N}$. When $B_1 \cup B_2$ cut across all weak paths between **S** and **R**, there do not exist vertices $u \in C$ and $v \in \mathbf{R}$-group, such that the edge (v, u) is in $\mathcal{N}$.

We prove the impossibility even for the best case where every other edge (other than those between C and R-group) exists and when every player knows the actual graph.

Define two executions $\mathbf{E}_1$ and $\mathbf{E}_2$ as follows. In both executions the vertices in $\mathbf{R}$-group hold the random inputs $\{\rho_u | u \in \mathbf{R}\text{-group }\}$. In the execution $\mathbf{E}_\alpha \in \{\mathbf{E}_1, \mathbf{E}_2\}$, the Byzantine set B_α is corrupt and the message m_α is transmitted by $\mathbf{S}$, the random inputs of the vertices in $(C \cup B_{\overline{\alpha}})$[2] are $\{\rho_u | u \in (C \cup B_{\overline{\alpha}})\}$. The behavior of the Byzantine set

B_α in the execution $\mathbf{E}_\alpha$ is to send no message whatsoever to $C \cup B_{\overline{\alpha}}$ and to send to $\mathbf{R}$-group exactly the same messages that are sent to $\mathbf{R}$-group by the honest B_α in the execution $\mathbf{E}_{\overline{\alpha}}$. In order for the Byzantine set B_α to behave as specified in the execution $\mathbf{E}_\alpha$, the adversary needs to simulate the behavior of $(C \cup B_\alpha)$ in the execution $\mathbf{E}_{\overline{\alpha}}$. To achieve this task, the adversary simulates round-by-round the behavior of the vertices in $(C \cup B_\alpha)$ for the execution $\mathbf{E}_{\overline{\alpha}}$ using $\{\rho_u | u \in (C \cup B_\alpha)\}$ as the random inputs for the vertices in $(C \cup B_\alpha)$. At the beginning of each round, each simulated player has a history of messages that it got in the simulation of the previous rounds and its simulated local random input. The simulated player sends during the simulation the same messages that the honest player would send in the original protocol in the same state. The simulated messages that (players in) B_α sends to $\mathbf{R}$ are really sent by the players. All other messages are used only to update the history for the next round. The messages which are added to the history of each simulated vertex are the real messages that are sent by players in $\mathbf{R}$-group and the simulated messages that are sent by the vertices in $(C \cup B_\alpha)$. No messages from $B_{\overline{\alpha}}$ are added to history. The history of messages of each simulated vertex in execution $\mathbf{E}_\alpha$ is the same as the history of the vertex in execution $\mathbf{E}_{\overline{\alpha}}$. Therefore, the messages sent by B_1 and B_2 to members of $\mathbf{R}$-group in both executions are exactly the same and the members of $\mathbf{R}$-group and in particular the receiver $\mathbf{R}$ receive and send the same messages in both executions. Thus, the receiver $\mathbf{R}$ cannot distinguish whether the set B_1 is corrupt and the message transmitted by $\mathbf{S}$ is m_1 or the set B_2 is corrupt and the message transmitted by $\mathbf{S}$ is m_2. Now, consider all the pairs of executions where the random inputs range over all possible values. In each pair of executions, whenever $\mathbf{R}$ accepts the correct message in one execution it commits an error in the other. Thus, for any strategy by $\mathbf{R}$ for choosing whether to receive m_1 or m_2 there is some α such that when m_α is transmitted, the receiver accepts m_α with probability at most $\frac{1}{2}$. ■

Lemma C.0.2 *Following Definition 11, if $\exists W$ such that every weak path p that avoids both B_1 and B_2 between $\mathbf{S}$ and $\mathbf{R}$ has a node, say w, that has both its adjacent edges (along p) directed inwards and $w \in W$ and both B_1 and B_2 are vertex cut-sets between w and $\mathbf{R}$, then a PRC protocol between $\mathbf{S}$ and $\mathbf{R}$ does not exist. In other words,if every strong path from w to $\mathbf{R}$ passes through both B_1 and B_2, then a PRC protocol between $\mathbf{S}$ and $\mathbf{R}$ does not exist.*

Proof: Note 2 gives us a glimpse of the state of the network $\mathcal{N}$ following Definition 11.

In Lemma 4.1.2, we proved that when there are no weak paths between $\mathbf{S}$ and $\mathbf{R}$ that avoid B_1 and B_2, PRC protocol does not exist. We now show that in spite of the presence of multiple such weak paths between $\mathbf{S}$ and $\mathbf{R}$ that avoid B_1 and B_2, if they have a node of the type w, with both its edges inwards towards w along the path, PRC protocol does not exist when both B_1 and B_2 are vertex cut-sets between w and $\mathbf{R}$. We take the case where every player complete knowledge of

[2] We denote $\overline{1} = 2$ and vice-versa;

the topology in spite of which PRC is shown to be impossible.

At least one edge from these weak paths must be from a node in **R**-group to another node in C (since these are paths outside $(B_1 \cup B_2)$ and from **S** to **R**). We will show that removing that edge has no effect on the possibility of PRC thereby proving the required result.

Firstly, how can these edges be useful? The answer is that they can be used by players in **R**-group to send some secret messages to the players in C such that the adversary, oblivious of these messages, cannot simulate the messages of C without being distinguished by **R**-group. However, if we are able to show that no such secret information can help PRC from **S** to **R**, then we are through. We do the same now.

A node x is said to have no influence on **R** if the output of **R** is independent of values sent by x. Otherwise x is said to influence **R**. Consider an edge (y, x) in $\mathcal{N}$ such that $y \in$ **R**-group and $x \in C$. We need to know whether x can influence **R** by using the data received from y. Suppose we manage to show that it cannot then we are through since what it means is that data sent along the edge (y, x) has no effect on **R** and hence can be ignored. We now proceed to prove the same.

Suppose that the node **R** can be influenced by x. This (at least) means that there must be a path $x, w_1, w_2, \ldots, w_q, \mathbf{R}$ in $\mathcal{N}$ such that x transmits some information to w_1, then w_1 transmits some information to w_2 that depends on the information it got from x and so on until some information gets to **R**.[3]

Given that every path from x to **R** passes through some node(s) in B_α *followed* by some node(s) in $B_{\overline{\alpha}}$ for some $\alpha \in \{1, 2\}$, the adversary if it corrupts the α^{th} set in $\mathcal{A} = \{B_1, B_2\}$, does the following: let w_j be the first vertex in B_α on a path from x to **R**. The corrupt w_j ignores the real messages that it gets from the players in $C \cup B_{\overline{\alpha}}$ and thus the messages that it sends do not depend on the message sent by x. Similarly, the messages sent by x when B_α simulates the players in C do not influence the messages it sends to **R** since the path from x to **R** passes through at least one vertex from $B_{\overline{\alpha}}$ and no messages are sent by players in $B_{\overline{\alpha}}$ during the simulation. Thus even if **R** may know that the correct secret (that was exchanged using the edge (y, x)) was not used, he will not know which set in $\mathcal{A}$ to blame. Thus the simulated messages of x have no influence on the messages received by **R** and can be ignored. Hence, the impossibility of PRC proved in Lemma 4.1.2 is not altered by using the edges from **R**-group to C. ■

D Sufficiency Proof of Theorem 4.1

D.1 Sufficiency Proof

For Sufficiency, Network $\mathcal{N}$ that is not in Critical Combination guarantees the existence of a PRC protocol from S to R in $\mathcal{N}$. So, we prove by giving a protocol and its proof of correctness.

Note 2 gives us the glimpse of the state of the network $\mathcal{N}$ following Definition 11.

[3] Since the network is synchronous, it may be possible to transmit information without actually sending message bits. However, even such transmissions are possible only between nodes that can actually exchange some message-bits as well. Thus, an information-path is necessarily a physical path too.

Since there are n players, and t can be corrupted, there are $\binom{|\mathbb{P}|}{t}$ options in front of the adversary,that is there are exactly $\binom{|\mathbb{P}|}{t}$ distinct ways of corrupting exactly t players. Let each of the $\binom{|\mathbb{P}|}{t}$ distinct subsets of size t be represented as $\{B_1, B_2, B_3, \ldots, B_{\binom{|\mathbb{P}|}{t}}\}$ where $B_i \subset \mathbb{P}$ and $|B_i| = t$. First, we show how to design a "PRC" sub-protocol assuming that the adversary is allowed to choose only from *two* of the $\binom{|\mathbb{P}|}{t}$ options that originally existed. In other words, we are only concerned about an adversary that may corrupt the players in the set B_α or the set B_β, where $1 \leq \alpha, \beta \leq \binom{|\mathbb{P}|}{t}$ and $\alpha \neq \beta$. Let us denote the resulting sub-protocol as $\Pi_{\alpha\beta}$. In the sequel, we show how to use all the sub-protocols $\Pi_{\alpha\beta}$ (there are clearly $\binom{\binom{|\mathbb{P}|}{t}}{2}$ of them) to design a grand protocol Π that can be proved to be the required PRC protocol. In the Definition 11, the two t-sized sets B_1 and B_2 can be understood as the sets of players that an adversary may corrupt in one of the instances of B_α and B_β.

An honest player is the player not corrupted by the adversary. An honest path is understood as the path that avoids the sets B_α and B_β. An honest player in the network in possession of the actual edge-set behaves differently from an honest player which has a doubleton set. When the player knows $\mathcal{E}$, all their communication, that is, sending and receiving messages is along the lines of edges in $\mathcal{E}$ only. When the player has a doubleton set as its $\mathcal{TK}_i$, it sends messages along both, the edges in actual edge-set $\mathcal{E}$ and the other edge-set in its $\mathcal{TK}_i$. It is never sure which of its message is valid, because all communication along the false edges is lost, and the identity of the false edges is not with the player. Now, an honest player with a doubleton set as its $\mathcal{TK}_i$ accepts all messages that it receives which it identifies as valid, that is, those belonging to the edges in any of the two edge-sets in its $\mathcal{TK}_i$. All players corrupted by the adversary, w.l.o.g.,can be assumed to know $\mathcal{TK}$ and the actual edge-set $\mathcal{E}$. This is a modest assumption when dealing with an adversary. An honest player drops all messages from a player if it identifies that it is corrupted by the adversary.

Designing the sub-protocol $\Pi_{\alpha\beta}$*:* Critical Combination does not occur in network $\mathcal{N}$ and this implies all that all the conditions that cause critical combination are falsified. We see the consequences of the same here, and use this to design our sub-protocol.

Neither of B_α or B_β cut across all strong paths between $\mathbf{S}$ and $\mathbf{R}$. Since B_α and B_β are the sets chosen by adversary one of which it can corrupt, there must be at least one honest strong path from $\mathbf{S}$ and $\mathbf{R}$ that does not pass through either B_α or B_β in $\mathcal{N}$.

The deletion of both the sets B_α and B_β from the network $\mathcal{N}$ does not cut across all weak paths between $\mathbf{S}$ and $\mathbf{R}$. There must exist at least one honest weak path from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$ that avoids both the sets B_α and B_β.

We start with this honest weak path, say p. We consider the following two cases in the design of the sub-protocol $\Pi_{\alpha\beta}$:

Case (1) :*The path p is such that $w = \mathbf{S}$:* In this case, the path p contains a player y (which may be $\mathbf{S}$ or $\mathbf{R}$ too) such that p is the combination of the strong path from y to $\mathbf{S}$ and the strong path from y to $\mathbf{R}$. In other words, $y \in$($\mathbf{S}$-group $\cap$$\mathbf{R}$-group). We know that $\mathbf{R}$-group has the edge-sets $\{E_0, E_1\}$ as its topology knowledge. If $\mathbf{S}$-group$\cap X \neq \emptyset$, then $\mathbf{S}$ would know the actual graph, else, it would have the same topology knowledge as $\mathbf{R}$-group, $\{E_0, E_1\}$. We give the protocol for the case where both $\mathbf{S}$ and $\mathbf{R}$ do not have the actual graph $\mathcal{N}$ and are in possession of $\{E_0, E_1\}$, one of which is known to be $\mathcal{N}$, as per the Definition 12. The other case is similar and follows the same approach. *case (i):* Notice that $y \in$ $\mathbf{R}$-group, so even y has $\{E_0, E_1\}$ as its topology knowledge.

Each of these edge-sets is such that each has a weak path that does not pass through the two sets B_α and B_β. Let the path along the actual edge-set (w.l.o.g, say E_0) be p and along E_1 be p'. In p', we have a y' which has a strong path from it to **S** and **R**. The state as defined in Note 2 is the same in both the edge-sets. Note that all players in p and p' are honest. The protocol that is run on one path p is correspondingly replicated on p'. We give the protocol for p: First, y sends to both **S** and **R**,along its both edge-sets, a message with two parts: one, a set of random keys, two, an array of signatures. Each player appends its signature to the second part of the message as it forwards the message to the next player in the path p. The random keys K_1, K_2 and K_3 (where K_i are elements of a random field $\mathbb{F}$, which in turn is also the message space), along with the list of signatures of the players that the messages have seen, is sent to both **S** and **R** along the path p. **S, R** receive two sets of the same three keys along the actual edge-set, and E_1 from y in p. Along p', suppose the random keys sent by y' be K'_1, K'_2 and K'_3 (where K'_i are elements of a random field $\mathbb{F}$, which in turn is also the message space) . If **S, R** receive these three keys along both the edge-sets E_0 and E_1, then they accept them as they cannot distinguish between the two edge-sets as to which is the correct one. **S,R** end up with two distinct sets of keys - (K'_1, K'_2 and K'_3) and (K_1, K_2 and K_3). Next,**S** computes two values: ψ, ψ_1; two signatures: χ,χ_1; where $\psi = (M + K_1)$, $\chi = (K_2(M + K_1) + K_3)$ and $\psi_1 = (M + K'_1)$, $\chi_1 = (K'_2(M + K'_1) + K'_3)$, and M is the message that needs to be reliably transmitted. **S** sends two messages in each edge-set E_0 and E_1 [4] to **R** along *all* the vertex-disjoint strong paths each containing: a value (ψ/ψ_1), a signature(χ/χ_1), and an array of signatures. Each player appends its signature to the array of signatures as it forwards the message to the next player in the path p or correspondingly in p'. Now, **R** receives two values - two each of ψ' and ψ'_1; two signatures - two each of χ' and χ'_1 along two different paths as are given in each of the edge-sets E_0 and E_1. Notice that, **R** has knowledge of (K_1, K_2 and K_3) and (K'_1, K'_2 and K'_3). Hence it can easily verify if $\chi' \overset{?}{=} K_2 * \psi' + K_3$ (correspondingly it verifies for χ'_1). **R** reacts as follows: If the received value ψ' has a valid signature ($\chi' = K_2 * \psi' + K_3$), then **R** outputs ($\psi' - K_1$) (correspondingly it outputs ($\psi'_1 - K'_1$) in the other edge-set); furthermore, among all the received values, at least one of them is guaranteed to be valid (because at least one honest strong path exists!). The probability that **R** outputs the same message in both the edge-sets is high,namely $1 - \frac{1}{|\mathbb{F}|}$,which can be made $(1 - \delta)$ by suitably choosing $\mathbb{F}$.

Case (2): *The path p is such that there are $k > 0$ players like w ($w \neq$ **S**), say $w_1, \ldots, w_k$ along p:* We will first consider the case when $k = 1$. For each of the subsequent cases ($k > 1$), we repeat the appropriate protocols given below on all w_i's ($1 \leq i \leq k$) and in the sequel succeed in establishing reliable communication between **S** and **R** with a high probability.

Since we start with the assumption that conditions on Definition 11 are falsified, note that every strong path from w to **R** does not pass through both B_α and B_β for a $w \in W$ that is on the honest weak path p from w_1 to **R**. That is, there must exist a strong path Q from w_1 to **R** that does not pass through nodes in either the set B_α or the set B_β.

Recall that p must contain a node y (which may be **R**) such that there is strong path from y to w_1 (along p) and there is a strong path from y to **R** (also along p). In other words, $y \in (w_1 - group \cap \mathbf{R} - group)$. We know that **R**-group has the edge-sets $\{E_0, E_1\}$ as its topology knowledge. If w_1-group $\cap$ $X \neq \emptyset$, then w_1 would know the actual graph, else, it would have the same topology knowledge as **R**-group, $\{E_0, E_1\}$. The protocol we give below is in two parts. Part I deals with the protocols to for communication between w_1 and **R** for each of the four cases given

[4] Note that **S** can verify if it has at least one honest strong path from it to **R** or not, and distinguish it with E_1

above. Part II deal with how, from Part I, we move on to ensure reliable communication between **S** and **R** with a very high probability.

Part I: *Protocols for communication between* w_1 *and* **R***: case (i):* We give the protocol for the case where both w_1 and **R** do not have the actual graph $\mathcal{N}$ and are in possession of $\{E_0, E_1\}$, one of which is known to be $\mathcal{N}$, as per the Definition 12. The other case is similar and follows the same approach. The protocol for w_1, y,**R** is similar to the case (i) in Case 1 above till each of w_1, **R** end up with two distinct sets of keys - (K'_1, K'_2 and K'_3) and (K_1, K_2 and K_3) just as **S, R** do.

Next, w_1 computes two values: ψ, ψ_1; two signatures: χ,χ_1; where $\psi = (M_{w_1} + K_1)$, $\chi = (K_2(M_{w_1} + K_1) + K_3)$ and $\psi_1 = (M_{w_1} + K'_1)$, $\chi_1 = (K'_2(M_{w_1} + K'_1) + K'_3)$, and M_{w_1} is the message from w_1 that is to be reliably transmitted to **R**. Let the path along the E_0 be Q and along E_1 be Q'. w_1 sends two messages in each edge-set (E_0 and E_1) to **R** along Q and Q' each containing: a value (ψ/ψ_1), a signature(χ/χ_1), and an array of signatures. Each player appends its signature to the array of signatures as it forwards the message to the next player in the path Q or correspondingly in Q'. Now, **R** receives two values - two each of ψ' and ψ'_1; two signatures - two each of χ' and χ'_1 along two different paths as are given in each of the edge-sets E_0 and E_1. **R** verifies using both the set of keys in its possession. Using these, it can easily verify if $\chi' \overset{?}{=} K_2 * \psi' + K_3$. It verifies on all combinations of ψ',ψ'_1,χ' and χ'_1. **R** reacts as follows: If the received value ψ' has a valid signature on at least one of the combination (say $\chi' = K_2 * \psi' + K_3$), then **R** outputs $(\psi' - K_1)$; else (that is if either the signature is invalid($\chi' \neq K_2 * \psi' + K_3$) or the original message is not received), **R** *knows* the identity (among the two possibilities of α or β) of the set that is the corrupt set.

Since we start with the assumption that conditions on Definition 11 are falsified, we note that $\forall$ nodes in $(W \cup (Z \cap X))$, for $i \in \{0, 1\}$, B_1 is not a vertex cut-set to **R** in the edge-set $E_i \in \mathcal{TK}_G$, and B_2 is not a vertex cut-set to **R** in the edge-set $E_{\overline{i}} \in \mathcal{TK}_G$, the path Q completely avoids the players from one of these sets say B_j, $j \in \{1, 2\}$; This clearly means that a faulty path Q (since a wrong message was delivered) entails that set $B_{\overline{j}}$ is corrupt (where $\overline{j} = \{1, 2\} - \{j\}$).

In Case (2) and its sub-cases above, if the set B_j, $j \in \{1, 2\}$, is not corrupt (which means that the other set may be corrupt), then **R** receives the correct message with certainty while the adversary has no information about the message.On the other hand, if the set B_j is corrupt, then though the adversary still has no information about the transmitted message, he has complete control over **R**'s output. **R**'s output could therefore either be a valid message or a null message with the knowledge that (any subset of) B_j is corrupt. But, if **R** receives a valid message, it is the correct message with a very high probability.

Protocols for the four cases in Part I aim at one of the following: (a) Simulating a direct edge (in other sense, having a strong path that passes only through honest players) between w_1 and **R** so that message from w_1 can be successfully communicated to **R** (or) (b) Simulation of the direct edge fails, and **R** identifies which of the two sets in B_α, B_β is corrupt.

Part II: If a protocol in Part I succeeds in (a) above, then depending on whether w_1 (and thereby all such w_i's) belongs to **S**-group or vice-versa, that is, $w_1 \in$ **S**-group or **S** $\in w_1$-group we have two cases. Note that, there will exist w_i's where neither $w_i \in$ **S**-group nor **S** $\in w_i$-group. It is precisely for this reason that we repeat the appropriate protocols given below on all w_i's ($1 \leq i \leq k$) so as to arrive at a case where one of these two cases occurs.

If $w_1 \in$ **S**-group, then, there exists a direct path from w_1 to **S**. From the protocol in Part I, there is a direct path from w_1 to **R**. That is, $w_1 \in$ **R**-group, which implies that $w_1 \in$ (**S**-group $\cap$ **R**-group). Notice that w_1 is similar to the player y in path p in Case (1) of the sub-protocol $\Pi_{\alpha\beta}$. So, in this case, we follow the protocol in Case (1) to establish reliable communication between **S** and **R**. If **S** $\in w_1$-group, then **S** sends the message that it wants to reliably communicate to **R** through the path between w_1 and **R**. Since the path is secure, and passes only through honest players, reliable communication takes place.

If a protocol in Part I succeeds in (b) above, then we do the following: Since there exists at least one honest strong path from **S** to **R**, it must avoid B_j. **S** sends the message along its both edge-sets which has two parts: the message M and an array of player indices. Each player appends its index to the array of player indices as it forwards the message to the next player along all these paths to **R**. The knowledge that B_j is corrupt is sufficient for **R** to recover the correct message passing through the honest path.

Note that this gives us the protocol for our uniform topology knowledge built with the help of $\mathcal{TK}_G$s. To give the protocol for a given 2-sized $\mathcal{TK}$, we use the means shown in the *If part* proof of the Theorem B.1. This completes our exercise of constructing the sub-protocol $\Pi_{\alpha\beta}$ that is guaranteed to work correctly only if one of B_α or B_β is chosen by the adversary.

Note that **R** can simulate the sub-protocol $\Pi_{\alpha\beta\gamma}$ which assumes that one among the *three* sets B_α or B_β or B_γ is chosen by the adversary. The simulation is done as follows: **R** takes the majority among the outputs of the three protocols $\Pi_{\alpha\beta}$, $\Pi_{\beta\gamma}$ and $\Pi_{\alpha\gamma}$. A majority is bound to exist since any set chosen by the adversary is tolerated in *two* of the three protocols. Next, R can simulate the sub-protocol which behaves like a PRC protocol as long as any one among a collection of *four* sets is chosen by the adversary. Continuing further, **R** will be able to simulate the protocol that behaves correctly if one among the collection of $\binom{n}{t}$ sets is chosen by the adversary. This protocol by definition is the PRC protocol from **S** to **R**! We conclude the sufficiency part of the proof. ■

E Corollaries

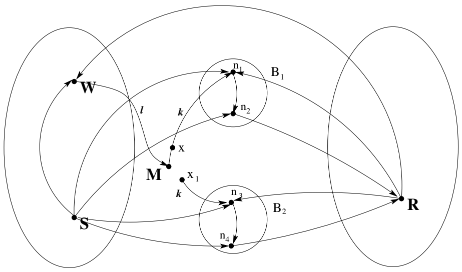

Figure 3: Network $\mathcal{N}$

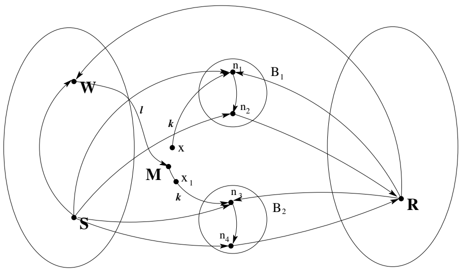

Figure 4: Network $\mathcal{N}'$

Corollary E.0.1 *There exist networks over which, a PRC protocol between a Sender* **S** *and the Receiver* **R** *in the network tolerating a t-adversary exists, if every player in the network has complete* *knowledge of the topology but does not exist if in the network, each player does not have knowledge of up to d_{min}-levels (hops) (both for in-edges as well as out-edges) where $\mathbf{d_{min}} = \lfloor \frac{n-2t}{3} \rfloor + 1$, where n is the total number of players in the network.*

Proof: We give a proof by construction. The Figure 3 represents the state of a network $\mathcal{N}$ similar to what is given in Note 2. The network $\mathcal{N}$ is constructed such that it satisfies the conditions for the possibility of PRC tolerating a t-adversary when each player has complete knowledge of the topology. Let there be n nodes in the network. The number of nodes in the path between W and M is l. Notice that the sets B_1 and B_2 cut across all strong paths between **S** and **R**. There are k nodes in the path connecting M to a node in B_1 and k nodes in a path from M to B_2 which is disconnected at node X_1. Sets B_1 and B_2 are sets of t nodes each, of which the adversary corrupts on of them. Both B_1 and B_2 are such that they contain nodes of the kind n_1, n_3 that form part of disjoint weak paths from **S** to **R**. They also contains nodes of the kind n_2, n_4 that form part of disjoint strong paths from **S** to **R**, totalling $t + 1$. There is a weak path between **S** and **R** passing through W. Notice that, in the network, **R** knows W, n_1, n_2, n_3, n_4 with knowledge of up to a single hop. **S** knows W,n_1,n_2,n_3,n_4 with a knowledge of up to a single hop. That is there are no nodes in between the paths from **S** and **R** to these nodes. Clearly, for this network, PRC protocol exists in the presence of a complete topology knowledge, because the underlying undirected graph is $(2t + 1)$-S,R connected and it has $(t + 1)$-strong paths for any number of nodes l in the path between W and M.

Let $l >= k - 1$ for the network. Clearly, for this network if we count the number of nodes and equate it to n, we get: $n = 4 + 2k + l + 2t$, which, when you take for the minimum value of l, $n = 2t + 3k + 3$.

In such a network $\mathcal{N}$, notice that the adversary can thwart the possibility of PRC in the absence of knowledge of $\mathbf{d_{min}}$ levels, because, for knowledge of up to $\mathbf{d_{min}} - 1$ hops, it can ensure indistinguishable views for the receiver over two executions, E_a and E_b by constructing another network similar to that of $\mathcal{N}$ - $\mathcal{N}'$ (see Figure 4). The adversarial strategy is exactly as described in the proof of Lemma 4.1.4, and it is as good as saying that $\mathcal{TK}_R$ contains both $\mathcal{N}$ and $\mathcal{N}'$. In the construction, it is only when each player has knowledge of up to $\mathbf{d_{min}}$ levels that **R** distinguishes between $\mathcal{N}$ and $\mathcal{N}'$. ■

Corollary E.0.2 *For any network, existence of a PRC protocol between* **S** *and* **R** *in the network tolerating a t-adversary when each player has complete knowledge of the topology implies existence of a PRC protocol between* **S** *and* **R** *in the network tolerating a t-adversary when each player has knowledge of up to $\mathbf{d_{min}}$-levels (hops) (both for in-edges as well as out-edges) where $\mathbf{d_{min}} = \lfloor \frac{n-2t}{3} \rfloor + 1$, where n is the total number of players in the network.*

Proof: We give an outline of how the proof proceeds. We first show that for the network constructed in the proof of Corollary 4.1.1, for the knowledge of up to $\mathbf{d_{min}}$-levels, a protocol exists for the solvability of PRC. We have already noted that $n = 2t + 3k + 3$, in the limiting case for l. Notice that one needs to have knowledge $d = k + 2$ levels to know whether node X_1 is connected to M or X is connected to M. Further, **R** would require $l + 2$ hops along **R**, W, $\ldots M$ path to reach M. Another hop is needed to know if M has an out-edge to X or X_1. When **R** knows d levels, it knows knowledge of both in-neighbours and out-neighbours up to level d, in the case when the minimum distance required to know as to which edge M is connected to, $l + 3 = k + 2$ giving us the limiting case for l. This gives us $\mathbf{d_{min}} = \lfloor \frac{n-2t}{3} \rfloor + 1$.

We now show that for any network over which PRC protocol for complete topology knowledge exists, knowledge of up to $\mathbf{d_{min}}$-levels is sufficient. We give a proof by contradiction. Suppose

$\mathbf{d_{min}}$-levels is not sufficient. Then the minimum D for which it would be sufficient would be $D = \mathbf{d_{min}} + 1$.

Notice that the condition given in Lemma 4.1.4 must be satisfied to experience the effect of topology knowledge on the possibility of PRC. The value of d_{min} is therefore dependent on the path-length of the paths between W and $\mathbf{R}$ in which there exists a node M where the paths diverge, one to B_1 in the first edge-set and the other to B_2 in the second edge-set in the topology knowledge of $\mathbf{R}$. Knowledge of up to that distance where the receiver $\mathbf{R}$ knows what this player M (which could be W also) is connected to, is what constitutes our quest.

The underlying undirected graph of network $\mathcal{N}$ is $(2t+1)$-**S,R** connected. In the directed graph, there are $(t+1)$-**S,R** vertex disjoint strong paths. If any of the vertex-disjoint strong paths is outside of B_1 and B_2, then **S** can reliably communicate to **R** using that honest strong path. Therefore, all strong paths must pass through either B_1 or B_2. The construction has a single directed edge from the sender **S** to a node in B_1 or B_2 for all the strong paths. Both B_1 and B_2 span across $2t$ weak paths between **S** and **R**, leaving one honest weak path outside of both B_1 and B_2, as is required by the characterization. Such a weak path would have a blocked node w with a strong path to **R** such that a node M where the paths diverge exists.

Supposing knowledge of up to $\mathbf{d_{min}}$-levels were not sufficient, we will count the number of nodes present in the network. If in a path between two nodes P and Q , there are k nodes, then the number of hops needed to reach Q from P is $k+1$. The path from **R**, W, $\ldots M$ must have $\mathbf{d_{min}}$ nodes, because only then, under our assumption that knowledge of up to D-levels is sufficient would hold because that would reveal whether M is connected to X or X_1 for knowledge of up to distance D. Notice that these d_{min} nodes include both M and W. Let there be l honest nodes in the weak path from a node in B_1 or B_2 through which **R** would know as to whether X is connected to M or X_1 is connected to M. The receiver would see up to a distance k in both edge-sets in its possession to reach B_1 and B_2. Further, we know that there are k nodes from M to B_1 and B_2. The number of nodes to be covered to reach X or X_1 through B_1 and B_2 respectively from **R** which includes the nodes k and l should be d_{min} because D would make sure that **R** knows whether X is connected to M or X_1 is connected to M. Now, if the nodes are counted along each of these paths, notice that along the paths through B_1 and B_2 to X and X_1, we include one node each from B_1 and B_2 in our calculations. So, the total number of nodes, which is n in the network, turns out to be, on assuming knowledge of up D levels $3 \times d_{min} + 2t - 2 + 2$. We have counted d_{min} nodes thrice in three paths, we counted two nodes, one each from B_1 and B_2, which we subtract, when we add the $2t$ nodes of the sets B_1 and B_2. We know the value of d_{min}, substituting which we get, $n = n + 3$, which is a contradiction. Therefore, our assumption that knowledge of up to D levels is sufficient is wrong.