Unconditionally Reliable Message Transmission in Directed Hypergraphs

Kannan Srinathan[2] **Arpita Patra**[1] **Ashish Choudhary**[1*] **C. Pandu Rangan**[1]

[1] Department of Computer Science and Engineering
Indian Institute of Technology Madras
Chennai India 600036
{arpita,ashishc}@cse.iitm.ernet.in, rangan@iitm.ernet.in
[2] Center for Security, Theory and Algorithmic Research
International Institute of Information Technology
Gachibowli, Hyderabad India
srinathan@iiit.ac.in

Abstract

We study the problem of *unconditionally reliable message transmission* (URMT), where two non-faulty players, the sender **S** and the receiver **R** are part of a synchronous network modeled as a directed hypergraph, a part of which may be under the influence of an adversary having *unbounded* computing power. **S** intends to transmit a message m to **R**, such that **R** should *correctly* obtain **S**'s message with probability at least $(1-\delta)$ for arbitrarily small $\delta > 0$. However, unlike most of the literature on this problem, we assume the adversary modeling the faults is **threshold mixed**, and can corrupt different set of nodes in Byzantine, passive and fail-stop fashion simultaneously. The main contribution of this work is the complete characterization of URMT in directed hypergraph tolerating such an adversary. Working out a direct characterization of URMT over directed hypergraphs tolerating threshold mixed adversary is highly un-intuitive. So we first propose a *novel* technique, which takes as input a *directed hypergraph* and a *threshold mixed adversary* on that hypergraph and outputs a corresponding *digraph*, along with a *non-threshold mixed adversary*, such that *URMT over the hypergraph tolerating the threshold mixed adversary is possible iff a special type of URMT is possible over the obtained digraph, tolerating the corresponding non-threshold mixed adversary.* We then characterize URMT in arbitrary digraphs tolerating *non-threshold mixed* adversary and modify it to obtain the characterization for *special type of URMT* over digraphs tolerating *non-threshold mixed adversary.* This completes the characterization of URMT over the original hypergraph. Surprisingly, our results indicate that even *passive* corruption, in collusion with active faults, substantially affects the reliability of URMT protocols! This is interesting because it is a general belief that passive corruption (eavesdropping) does not affect reliable communication.

1 Introduction

Consider a synchronous network, modeled as a directed hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$ where $\mathcal{P}$ is the set of nodes and $\mathcal{E} \subset \mathcal{P} \times 2^{\mathcal{P}}$ is the set of directed hyperedges. Some of the nodes in the network $\mathcal{D}$ are controlled by an adaptive[1] *threshold mixed* adversary $\mathcal{A}_{(t_b,t_p,t_f)}$ which possesses *unbounded* computing power and can corrupt disjoint set of t_b, t_p and t_f nodes in Byzantine, passive and fail-stop fashion respectively. In *Unconditionally Reliable Message Transmission* (URMT) problem over $\mathcal{D}$, a sender $\mathbf{S} \in \mathcal{P}$ wishes to send a message m, chosen from a finite field $\mathbb{F}$ to the receiver $\mathbf{R} \in \mathcal{P}$ (we assume that **S** and **R** are non-faulty), such that **R** recovers m with probability at least $(1-\delta)$, for arbitrarily small $\delta > 0$. If $\delta = 0$, then the problem is called *perfectly reliable message transmission* (PRMT), where **R** should output m with probability one. Since URMT problem assumes the adversary to have unbounded computing power, it

*Financial Support from Infosys Technology India Acknowledged.

[1] An adaptive adversary corrupts the nodes dynamically during the protocol execution. The nodes to be corrupted may depend upon the information obtained by the adversary so far during the protocol execution.

cannot be solved using public key cryptography, hash functions, digital signatures, etc which are based on the assumption that adversary has *bounded* computing power. Directed hypergraph is the most generic network model with the facility of multicasting. In certain scenarios, private one-to-one channels may not exist. Typical examples include Radio transmission and LAN network. Also in many practical scenarios, a base station can broadcast to a set of receivers, but the other way around communication might not be possible. In these cases, directed hypergraph is the only way to model the network.

One of the largest "consumers" of URMT protocols is the field of *distributed randomized algorithms* [8]. Unlike the need to simulate an error free communication channel between a mutually distant pair of nodes in a network, an important basic primitive in any distributed randomized algorithm is URMT protocol. Intuitively, the allowance of a small probability of error in the transmission should result in improvements in the fault tolerance of reliable message transmission protocols. What exactly is the improvement? We answer this question, considering the most generic network model namely directed hypergraph. The most natural, intriguing and interesting questions that can be posed in the context of URMT over directed hypergraphs are: (a) POSSIBILITY: What is the necessary and sufficient condition that a given directed hypergraph $\mathcal{D}$ should satisfy for the possibility of URMT from $\mathbf{S}$ to $\mathbf{R}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$? (b) Does allowing a small probability of error in the reliability of message transmission improves fault tolerance in the network? (c) A directed hyperedge may be visualized as "bunching up" a set of directed (simple) edges. So given a directed hypergraph $\mathcal{D}$, a digraph $\mathcal{G}$ can be obtained from $\mathcal{D}$ by replacing each hyperedge with the corresponding "bunch" of directed edges. Now the natural question is: Does there exists a directed hypergraph $\mathcal{D}$ such that URMT is possible over $\mathcal{D}$ but impossible over $\mathcal{G}$ tolerating $\mathcal{A}_{(t_b,t_f,t_p)}$? More precisely, do hyperedges possess some special property/power in the context of URMT? While the first question is the heart of this paper and answered in the next sections, the next two questions posed above are answered affirmatively little later in this section itself (subsection 1.3).

1.1 Existing work

The problem of URMT was first defined and solved by Franklin et. al. [5] over undirected graphs, tolerating *threshold Byzantine* adversary $\mathcal{A}_{t_b}$. As one of the key results, they proved that over undirected graphs, URMT tolerating $\mathcal{A}_{t_b}$ is possible iff PRMT (which is URMT with $\delta = 0$) tolerating $\mathcal{A}_{t_b}$ is possible. Subsequent works on URMT include [12, 2, 3]. Most recently, Srinathan et.al. [11] and Shankar et.al [10] has given the complete characterization for URMT in an arbitrary directed graph tolerating non-threshold and threshold **Byzantine adversary** respectively.

Considering hypergraph as underlying network model is not quite common in literature. The problem of secure communication against $\mathcal{A}_{t_p}$ in directed hypergraphs has been studied by Franklin et. al. [6]. Later Desmedt et.al. [2] has given the characterization for PRMT over hypergraphs tolerating $\mathcal{A}_{t_b}$.

1.2 Hypergraph Network Model and Threshold Mixed Adversary

In this work, we follow the hypergraph network model as in [6]. A directed hypergraph is denoted by $\mathcal{D} = (\mathcal{P}, \mathcal{E})$. A typical directed hyperedge e can be written as $e = (v, S)$, where $v \in \mathcal{P}$, $S \subseteq (\mathcal{P} \setminus v)$. We call v as the source node and the nodes in S as the destination nodes of e. The hyperedge e enables node v to send message to the nodes in S, identically. Even if v is corrupted and modifies (stops) the information passing through it, every node in S identically receive the modified information (no information). The hyperedge e is directed since only v can send messages to the nodes in S but the nodes in S cannot communicate among themselves or to v using e. However a node v_a in S can communicate with some other node v_b in S through other hyperedge in $\mathcal{E}$, with v_a as the source and v_b as one of the destination nodes. Also a node can be source/destination of many hyperedges. The hyperedge e is secure in the sense that any node outside the set $S \cup \{v\}$ learns nothing about the information sent over e.

We consider an adaptive *threshold mixed adversary* $\mathcal{A}_{(t_b,t_p,t_f)}$, who possesses *unbounded* computing power and controls at most t_b, t_p and t_f nodes in $\mathcal{D}$ in Byzantine, passive and fail-stop fashion respectively. The choice of the nodes which are under the control of $\mathcal{A}_{(t_b,t_p,t_f)}$ is decided dynamically during the protocol execution and may depend upon the information obtained by the adversary so far. Once a node is corrupted, it remains so throughout the protocol. If a node P is fail-stop corrupted by $\mathcal{A}_{(t_b,t_p,t_f)}$, then the adversary can force P to *crash* at will at any time during the execution of the protocol but can

not access its internal data and can not force its behaviour to deviate from the protocol. So till P is alive, it honestly follows the protocol. Also once P is crashed, it never becomes alive again. If a node P is passively corrupted by $\mathcal{A}_{(t_b,t_p,t_f)}$, then P honestly follows the protocol but the adversary has full access to internal data of P. If a node P is Byzantine corrupted by $\mathcal{A}_{(t_b,t_p,t_f)}$, then the adversary has full access to the internal data of P and can force P to deviate from the protocol arbitrarily. We assume that $\mathcal{A}_{(t_b,t_p,t_f)}$ acts in a "centralized" fashion and colludes among different corrupted nodes through "back channels". So, $\mathcal{A}_{(t_b,t_p,t_f)}$ can listen information from at most $(t_b + t_p)$ nodes and can pool all the information observed/obtained at these $t_b + t_p$ nodes in any manner in its local computation. The adversary can stop communication through at most $t_b + t_f$ nodes and can change information passing though at most t_b nodes.

Why to Study Mixed Adversary: In a typical large network, certain nodes may be strongly protected and few others may be moderately/weakly protected. An adversary may fail-stop(/eavesdrop in) a strongly protected node, while he may affect in a Byzantine fashion a weakly protected node. Thus, we may capture the abilities of an adversary in a more realistic manner using three parameters t_b, t_p, t_f where t_b, t_p, t_f are the number of nodes under the influence of Byzantine, passive and fail-stop adversary, respectively. Also it is better to grade different kinds of disruption done by adversary and consider them separately rather than treating every kind of fault as Byzantine fault as this is an "overkill". Even in practical scenario, when a hacker takes control of a router, it can disrupt the communication in variety of ways. Thus, we need to model the adversary as a mixed adversary.

1.3 Our Contribution, Significance and Impact

In this paper, we completely characterize URMT over an arbitrary directed hypergraph $\mathcal{D}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$. Working out a direct characterization of URMT over $\mathcal{D}$, tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$ is highly unintuitive. Hence, we follow the following framework to solve the problem:

Contribution 1 *We propose a method which takes a directed hypergraph $\mathcal{D}$, along with $\mathcal{A}_{(t_b,t_p,t_f)}$ and outputs a corresponding digraph $\mathcal{D}_{under}$, which we call as "underlying digraph" of $\mathcal{D}$, along with a non-threshold mixed adversary $\mathbb{A}_{under}$, such that URMT over $\mathcal{D}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$ is possible iff there exists a "special type of URMT" protocol in $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$. So to obtain characterization of URMT over directed hypergraph tolerating a threshold mixed adversary, we require the characterization for a "special type of URMT" in the "underlying digraph" tolerating a non-threshold mixed adversary.*

Contribution 2 *We then characterize URMT in $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$ and give modifications on it to arrive at the characterization of special type of URMT on $\mathcal{D}_{under}$. This along with Contribution 1 completes the characterization of URMT on $\mathcal{D}$, tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$.*

In the sequel we provide affirmative answers to all the following questions in order by demonstrating examples: (a) Does randomization helps in more fault tolerance? (b) Does hyperedges helps in the possibility of URMT? (c) Does passive corruption affects reliable communication? Consider the hypergraph $\mathcal{D}$ in Fig. 1 under the influence of $\mathcal{A}_{(1,0,0)}$, where $t_b = 1$ and $t_p = t_f = 0$. We can say the following regarding $\mathcal{D}$ which essentially shows the power of randomization:

- From [2], PRMT (URMT with $\delta = 0$) over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$ is impossible.
- URMT over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$ is possible and feasible.

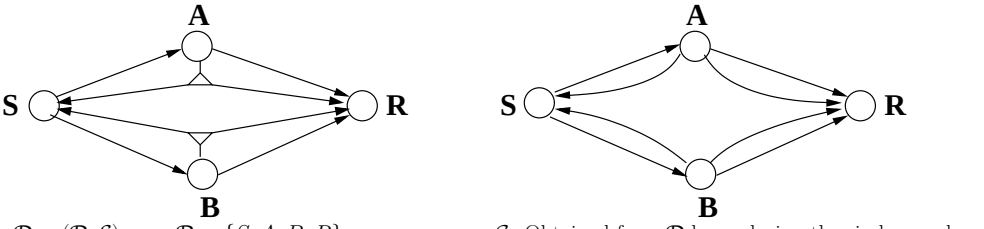
Claim 1 *URMT over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$ is possible and feasible.*

Proof: Consider the following protocol over $\mathcal{D}$. A selects three random values (keys) from finite field $\mathbb{F}$, say $K_1, K_2,$ and K_3. A sends the keys to $\mathbf{S}$ and $\mathbf{R}$ through the hyperedge $(A, \{\mathbf{S}, \mathbf{R}\})$. Since $(A, \{\mathbf{S}, \mathbf{R}\})$ is a hyperedge, both $\mathbf{S}$ and $\mathbf{R}$ receive the same set of keys or get nothing. If $\mathbf{S}$ receives the keys, he authenticates the message m by computing a two tuple $(K_1 + m, K_2(K_1 + m) + K_3)$ and sends it through the path $(\mathbf{S}, B, \mathbf{R})$. $\mathbf{S}$ also sends the message m over both the paths $(\mathbf{S}, B, \mathbf{R})$ and $(\mathbf{S}, A, \mathbf{R})$. Now if $\mathbf{R}$ does not receive any key from A, he detect A to be faulty and accept the message from path $(\mathbf{S}, B, \mathbf{R})$.

Otherwise, $\mathbf{R}$ receives the authenticated tuple say (c, d) over path $(\mathbf{S}, B, \mathbf{R})$ and the keys from A with which $\mathbf{R}$ checks $d \stackrel{?}{=} K_2 c + K_3$. If the test passes he takes $c - K_1$ as the message, else $\mathbf{R}$ knows node B is faulty and accept the message from path $(\mathbf{S}, A, \mathbf{R})$. The proof of correctness of the protocol is similar to the information checking protocol of [9]. It can be shown that except with probability $\delta = \frac{1}{|\mathbb{F}|}$, $\mathbf{R}$ outputs $m' = m$. This is so because B does not know the keys. So the only way he can forge an incorrect message is to correctly guess K_2 or K_3. Now by setting the field size $|\mathbb{F}|$ to be arbitrarily large, we can reduce the error probability to an arbitrarily small quantity. □

Next we show that unlike PRMT, *hyperedges do help in the possibility of URMT.* A very straight-forward implication that can be drawn from the characterization of PRMT in directed hypergraph against $\mathcal{A}_{t_b}$, stated in [2], is that, replacing every hyperedge of a hypergraph by a collection of underlying simple directed edges does not affect the possibility of PRMT over the hypergraph. This means there does not exist a hypergraph such that PRMT is possible on it but is impossible in the digraph obtained from the hypergraph. However consider the hypergraph $\mathcal{D}$ in Figure 1 and its corresponding directed graph $\mathcal{G}$, which is obtained from $\mathcal{D}$ by replacing each hyperedge by its underlying simple directed edges. From Claim 1, URMT is possible in $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$. However, from [10], URMT is impossible in $\mathcal{G}$ tolerating $\mathcal{A}_{(1,0,0)}$. *Thus we conclude that hyperedges do help in the possibility of URMT but not PRMT.*

One of the long-standing and intuitive belief is that "passive corruption does not affect reliable communication". We contradict the belief by an example for URMT in directed hypergraph. In Figure 1, let $\mathcal{D}$ be under the control of $\mathcal{A}_{(1,1,0)}$, where $t_b = t_p = 1$ and $t_f = 0$. Let A be passive corrupted and B be Byzantine corrupted; then the protocol in Claim 1 will not work. The reason is that adversary always gets the keys which A sends to $\mathbf{S}$ and $\mathbf{R}$ (by eavesdropping A). Specifically $\mathcal{A}_{(1,1,0)}$ can use the information he eavesdrop at A, to corrupt the values appropriately at B, in such a way that the tuple along the path $(\mathbf{S}, B, \mathbf{R})$ passes the authentication test. Thus an adversary can *very effectively use the information obtained from the passively corrupted nodes at the nodes which he is controlling in a Byzantine fashion and affect the reliability of the protocol.*



$\mathcal{D} = (\mathcal{P}, \mathcal{E})$ $\quad \mathcal{P} = \{S, A, B, R\}$
$\mathcal{E} = \{(S, \{A\}), (A, \{R\}), (A, \{S, R\}, (S, \{B\}), (B, \{R\}), (B, \{S, R\})\}$

$\mathcal{G}$: Obtained from $\mathcal{D}$ by replacing the six hyperedges by corresponding directed edges.

Figure 1: Example of network illustrating the power of hyperedges with respect to URMT.

1.4 Digraph Network Model and Non-threshold Adversary

We now give few definitions related to digraph network model and non-threshold adversary, which are used in subsequent sections. A directed network is modeled as a digraph $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ where $\mathbb{P}$ is the set of nodes and $\mathbb{E}$ denotes the set of arcs in the digraph. The network is assumed to be synchronous, that is, the protocol is executed in a sequence of *rounds* wherein in each round, a node can send messages to it's out-neighbors, receive the messages sent in that round by it's in-neighbors and perform some computation on the received messages, in that order. A non-threshold adversary structure is an enumeration of all the possible "snapshots" of faults in the network. A single snapshot can be described by an ordered triple (B, E, F), where $B, E, F \subseteq \mathbb{P}$ and B, E and F are pairwise disjoint, and denotes the set of Byzantine, passive and fail-stop corrupted nodes. Thus, an adversary structure is a collection of such triples. More precisely, we denote the adversary structure by $\mathbb{A}$, where $\mathbb{A} \subseteq 2^{\mathbb{P} \times \mathbb{P} \times \mathbb{P}}$. The adversary structure is *monotone* in the sense that if $(B_1, E_1, F_1) \in \mathbb{A}$, then $\forall (B_2, E_2, F_2)$ such that $B_2 \subseteq B_1$ and $E_2 \subseteq E_1$ and $F_2 \subseteq F_1$, $(B_2, E_2, F_2) \in \mathbb{A}$. During the execution of the protocol, any one set from $\mathbb{A}$ would be active and the nodes from that set will be under the control of the adversary throughout the protocol execution. A threshold mixed adversary $\mathcal{A}_{(t_b, t_p, t_f)}$ is a special type of non-threshold adversary where size of B, E and F are bounded by t_b, t_p and t_f respectively. Note that $\mathbb{A}$ can be uniquely represented by listing the elements in its *maximal basis* which we define below:

Definition 1 (Maximal Basis of $\mathbb{A}$) *For any monotone adversary structure $\mathbb{A}$, its maximal basis $\bar{\mathbb{A}}$ is defined as $\bar{\mathbb{A}} = \{(B,E,F)|(B,E,F) \in \mathbb{A}$, and $\nexists (X,Y,Z) \in \mathbb{A}$ such that $(X,Y,Z) \neq (B,E,F)$ where $X \supseteq B, Y \supseteq E$ and $Z \supseteq F\}$.*

Definition 2 (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 digraph $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ if for each $1 \leq i < k$, $(v_i, v_{i+1}) \in \mathbb{E}$. We assume that (v_i,v_i) is a strong path from v_i to itself.*

Definition 3 (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 digraph $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ if there exists j, $1 \leq j \leq k$ such that the sequence v_j to v_1 as well as the sequence v_j to v_k are both strong paths in the network. Vertex v_j is called the* **head** *of the semi-strong path. Any strong path can be viewed as a semi-strong path. For example, the path $(\mathbf{S}, X^{Vir}_{e_5}, \mathbf{R})$ in graph $\mathcal{D}_{under}$ in Fig. 2 is a semi-strong path between $\mathbf{S}$ and $\mathbf{R}$, where $X^{Vir}_{e_5}$ is the head.*

Definition 4 (Authentication Function) *Let $\mathbb{F}$ be a finite field, $\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3 \in \mathbb{F} - \{0\}$ be three random keys and $m \in \mathbb{F}$ be a message. Then $auth(m, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3) = (\mathcal{K}_1 + m, \mathcal{K}_2(\mathcal{K}_1 + m) + \mathcal{K}_3)$.*

Suppose a random triplet $(\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3) \in \mathbb{F}^3 - \{(0,0,0)\}$ is correctly established between $\mathbf{S}$ and $\mathbf{R}$. For a message m, let $\mathbf{S}$ computes $auth(m, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ and sends it to $\mathbf{R}$ through a strong path, over which some of the nodes could be under the control of the adversary. If the adversary does not know $m, \mathcal{K}_1, \mathcal{K}_3$ and $\mathcal{K}_3$ in advance, then *auth* satisfies the following two important properties: (a) Even if adversary learns $auth(m, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$, m will remain information theoretic secure. (b) If the adversary changes $auth(m, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ to some other value, then except with an error probability of at most $\frac{1}{|\mathbb{F}|}$, $\mathbf{R}$ will be able to detect it. The proof of both the properties is similar to the proof of information checking protocol of [9] and hence is omitted.

2 Characterization for URMT in Directed Hypergraph

Here we characterize URMT on arbitrary directed hypergraphs tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$.

Definition 5 (Underlying Digraph) *Given a directed hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$ we define the underlying digraph $\mathcal{D}_{under} = (\mathcal{P}', \mathcal{E}')$ of the hypergraph $\mathcal{D}$ as follows: $\mathcal{P}' = (\mathcal{P} \cup \mathcal{V})$ is the set of nodes (real nodes and virtual nodes). We replace each hyperedge $e = (v, \{v_{j_1}, v_{j_2}, \ldots, v_{j_\alpha}\}) \in \mathcal{E}$, $\alpha \geq 1$, with a virtual player $X^{Vir}_e \in \mathcal{V}$ and the arcs (v, X^{Vir}_e), (X^{Vir}_e, v_{j_1}), (X^{Vir}_e, v_{j_2}), $\ldots$, and $(X^{Vir}_e, v_{j_\alpha})$ in $\mathcal{E}'$. Thus, $|\mathcal{V}|$ is the number of hyperedges in $\mathcal{D}$. It is clear that every virtual player has an unique in-neighbor which belongs to $\mathcal{P}$. Also the out-neighbors of each virtual player belongs to only $\mathcal{P}$. Note that two virtual players can never be adjacent to each other.*

A hypergraph, along with its corresponding digraph D_{under} is given in Fig. 2. The intuition behind our definition is that for every hypergraph $\mathcal{D}$ influenced by a threshold mixed adversary $\mathcal{A}_{(t_b,t_p,t_f)}$, there exists a corresponding digraph $\mathcal{D}_{under}$ and non-threshold mixed adversary $\mathbb{A}_{under}$, such that URMT over $\mathcal{D}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$ is possible iff a "special type of URMT" over $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$ is possible. We begin by defining this special kind of URMT protocol, denoted as $URMT_{special}$.

Definition 6 ($URMT_{special}$) *A URMT protocol over a digraph $\mathcal{D}_{under} = (\mathcal{P} \cup \mathcal{V}, \mathcal{E}')$ (which is the underlying digraph of the directed hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$) is called as special, denoted by $URMT_{special}$, if in that protocol, the programs delegated to each of the virtual players (i.e., the players in $\mathcal{V}$) is known to all and is deterministic i.e., the players in $\mathcal{V}$ do not use any internal random coins in the protocol.*

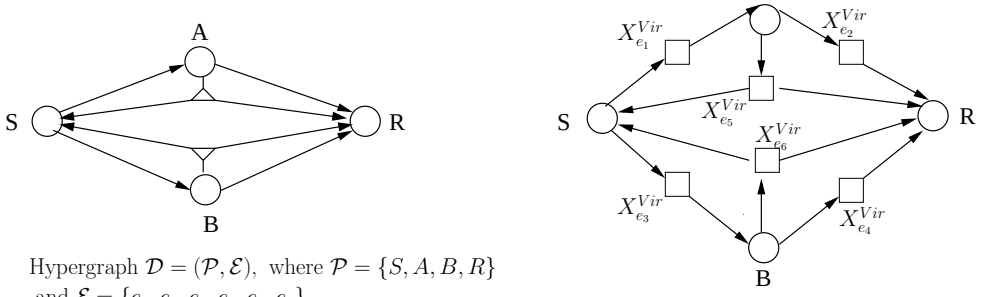
Definition 7 ($\mathbb{A}_{under}$) *Let $\mathcal{D} = (\mathcal{P}, \mathcal{E})$ be a arbitrary directed hypergraph under the influence of $\mathcal{A}_{(t_b,t_p,t_f)}$. Also let $\mathcal{D}_{under} = (\mathcal{P}', \mathcal{E}')$ be the underlying digraph of $\mathcal{D}$. The non-threshold mixed adversary $\mathbb{A}_{under}$ over $\mathcal{D}_{under}$, corresponding to $\mathcal{A}_{(t_b,t_p,t_f)}$ in $\mathcal{D}$ is defined as:*

$$\mathbb{A}_{under} = \left\{ (B,E,F) \left| \begin{array}{l} |B \cap \mathcal{P}| \leq t_b,\ (B \cap \mathcal{V}) = \emptyset,\ |F \cap \mathcal{P}| \leq t_f,\ (F \cap \mathcal{V}) = \emptyset, \\ E = E_{real} \cup E_{vir}, |E_{real} \cap \mathcal{P}| \leq t_p, (E_{vir} \cap \mathcal{P}) = \emptyset, \\ E_{vir} = \left\{ \nu \in \mathcal{V} \left| \begin{array}{l} \text{there exists a player } x \in (B \cup E_{real}) \text{ such that either } (x,\nu) \text{ or } (\nu,x) \text{ is} \\ \text{an arc in } \mathcal{D}_{under} \end{array} \right. \right\} \end{array} \right. \right\}$$

From the definition of $\mathbb{A}_{under}$, it is clear that every disjoint selection of t_b, t_p and t_f nodes from the set of nodes in $\mathcal{D}$ corresponds to an element in $\mathbb{A}_{under}$. To construct an element $(B, E, F) \in \mathbb{A}_{under}$, we first select a *possible* combination of disjoint set of t_b, t_p and t_f nodes from $\mathcal{P}$ (set of physical nodes) and assign them to B, E_{real} and F respectively. Now E_{vir} is constructed by adding the virtual nodes X_e^{Vir} corresponding to hyperedge $e = (v, \{v_{j_1}, v_{j_2}, \ldots, v_{j_\alpha}\})$ such that at least one of the nodes $v, v_{j_1}, v_{j_2}, \ldots, v_{j_\alpha} \in (B \cup E_{real})$. Finally $E = E_{real} \cup E_{vir}$. Modeling the hypergraph $\mathcal{D}$ with directed graph $\mathcal{D}_{under}$ and extending the adversary to $\mathbb{A}_{under}$ allows to work on a comparatively simpler digraph model. Consider the hypergraph $\mathcal{D}$ shown in Fig. 2 under the influence of $\mathcal{A}_{(1,1,0)}$. The corresponding $\mathbb{A}_{under}$ in $\mathcal{D}_{under}$ (shown in the same figure) is represented by its maximal basis $\overline{\mathbb{A}}_{under} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$ where $(B_1, E_1, F_1) = (\{A\}, \{B, X_{e_1}^{Vir}, X_{e_1}^{Vir}, X_{e_2}^{Vir}, X_{e_3}^{Vir}, X_{e_4}^{Vir}, X_{e_5}^{Vir}, X_{e_6}^{Vir}\}, \emptyset)$ and $(B_2, E_2, F_2) = (\{B\}, \{A, X_{e_1}^{Vir}, X_{e_1}^{Vir}, X_{e_2}^{Vir}, X_{e_3}^{Vir}, X_{e_4}^{Vir}, X_{e_5}^{Vir}, X_{e_6}^{Vir}\}, \emptyset)$.

Theorem 1 *A URMT protocol from* **S** *to* **R** *in a directed hypergraph* $\mathcal{D} = (\mathcal{P}, \mathcal{E})$ *tolerating* $\mathcal{A}_{(t_b,t_p,t_f)}$ *is possible iff* $URMT_{special}$ *from* **S** *to* **R** *over* $\mathcal{D}_{under}$ *is possible tolerating* $\mathbb{A}_{under}$.

PROOF: If part: Let Π' be a $URMT_{special}$ protocol from **S** to **R** over $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$. We now construct a URMT protocol Π in $\mathcal{D}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$, using Π'. In protocol Π', the virtual players run a deterministic program. Thus, if in Π' some message m is sent by $v \in \mathcal{P}$ to $X_e^{Vir} \in \mathcal{V}$ who then forwards the respective outputs to all his out-neighbors, the same may be exactly simulated by v just using m and the code of X_e^{Vir} to compute the outputs of all the out-neighbors of X_e^{Vir}, say $\mu_1, \mu_2, \ldots, \mu_k$ (if there are k out-neighbors of X_e^{Vir}) and respectively sending μ_j to the j^{th} out-neighbor routed through X_e^{Vir} (this kind of simulation is possible since the code run by X_e^{Vir} is deterministic and known to all). This in turn is equivalent to the real player v sending all the respective outputs $\mu_1, \mu_2, \ldots, \mu_k$ to all the out-neighbors via the hyperedge in $\mathcal{D}$ and each out-neighbor picking-up only what is due to him. Note that this step works because of the way in which we have defined the adversary structure $\mathbb{A}_{under}$ — we said that if adversary can read the memory of either the source node or one of the destination nodes of $e = (v, \{v_{j_1}, v_{j_2}, \ldots, v_{j_k}\})$, then the adversary can also read X_e^{Vir}'s memory itself. Consequently, we may assume that all data in X_e^{Vir}'s memory may be safely sent to all his out-neighbors without affecting the correctness of the simulation. It is evident that the view of the adversary as well as the out-neighbors is the same in both the original and the simulated versions. Thus, we can easily design URMT protocol Π.



Hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$, where $\mathcal{P} = \{S, A, B, R\}$ and $\mathcal{E} = \{e_1, e_2, e_3, e_4, e_5, e_6\}$ with $e_1 = (S, \{A\})$, $e_2 = (A, \{R\})$, $e_3 = (S, \{B\})$, $e_4 = (B, \{R\})$, $e_5 = (A, \{S, R\})$, $e_6 = (B, \{S, R\})$

$\mathcal{D}_{under} = (\mathcal{P}', \mathcal{E}'), \mathcal{P}' = \mathcal{P} \cup \mathcal{V}$, $\mathcal{V} = \{X_{e_1}^{Vir}, X_{e_2}^{Vir}, X_{e_3}^{Vir}, X_{e_4}^{Vir}, X_{e_5}^{Vir}, X_{e_6}^{Vir}\}$
□ denotes virtual nodes from the set $\mathcal{V}$

Figure 2: A directed hypergraph $\mathcal{D}$ and it's corresponding $\mathcal{D}_{under}$

Only if part: This is much easier. Suppose there exists a URMT protocol Π in the directed hypergraph $\mathcal{D}$. We now show that a $URMT_{special}$ protocol Π' in the digraph $\mathcal{D}_{under}$ exists. This can be seen as follows — we simulate a send of a value m along a hyperedge $e = (v, \{\{v_{j_1}, v_{j_2}, \ldots, v_{j_k}\})$ in the protocol Π over $\mathcal{D}$ by sending the value m first from v to the virtual player corresponding to the hyperedge e, namely, X_e^{Vir}, who then forwards it to the receivers v_{j_1}, v_{j_2}, …, and v_{j_k} in the protocol Π' over the network $\mathcal{D}_{under}$. Hence the theorem holds. □

So according to Theorem 1, our next concern is to characterize $URMT_{special}$ over $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$. For that, in the next section, we first characterize URMT in arbitrary digraphs tolerating non-threshold mixed adversary. This gives the necessary and sufficient condition for the existence of a *URMT protocol* over $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$. Now, the only reason why a URMT protocol may exist in $\mathcal{D}_{under}$ but a $URMT_{special}$ protocol does not exist is that in $\mathcal{D}_{under}$ there are some virtual players which can not act as a physical node (e.g. can not do random coin toss etc.). The modification required to obtain the

characterization of $URMT_{special}$ in arbitrary digraphs ($\mathcal{D}_{under}$) from the characterization of URMT in arbitrary digraphs ($\mathcal{D}_{under}$) is described in section 7. This will complete the characterization of URMT in $\mathcal{D}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$.

3 URMT in Digraphs Tolerating Non-Threshold Adversary

We now characterize URMT in an arbitrary synchronous directed graph $\mathcal{N}$ tolerating an arbitrary non-threshold adversary $\mathbb{A}$. Working out a direct characterization of URMT tolerating entire $\mathbb{A}$ is highly complex and non-intuitive. Rather it is easy to think of a characterization tolerating small sized subsets from $\mathbb{A}$. We now show that URMT tolerating an arbitrary non-threshold adversary $\mathbb{A}$ is possible iff URMT is possible tolerating every subset $\mathcal{A}$ of $\mathbb{A}$, with maximal basis $\bar{\mathcal{A}}$ of size two.

Theorem 2 *URMT in a digraph $\mathcal{N}$ tolerating a non-threshold adversary $\mathbb{A}$ is possible iff URMT is possible in $\mathcal{N}$ tolerating any $\mathcal{A} \subseteq \mathbb{A}$ with maximal basis $\bar{\mathcal{A}}$ of size two.*

PROOF (SKETCH): Necessity is obvious. For sufficiency, we show that if URMT is possible in $\mathcal{N}$ tolerating any $\mathcal{A} \subseteq \mathbb{A}$ with maximal basis $\bar{\mathcal{A}}$ of size two, then URMT is also possible in $\mathcal{N}$ tolerating any $\mathcal{A} \subseteq \mathbb{A}$ with maximal basis of size three. Then using induction, we show that it is possible to design URMT in $\mathcal{N}$ which tolerates the entire adversary structure $\mathbb{A}$. For complete proof, see **APPENDIX A**. □

Remark 1 *The protocol given in Theorem 2 is an inductive protocol and is exponential in the size of $\mathbb{A}$. Designing an efficient URMT protocol tolerating $\mathbb{A}$ is left as an open problem.*

Theorem 2 shows that in order to get a complete characterization of URMT tolerating the entire adversary structure $\mathbb{A}$, it is enough if we characterize URMT tolerating every $\mathcal{A} \subseteq \mathbb{A}$ with maximal basis $\bar{\mathcal{A}}$ of size two. This is our main concern in the rest of the paper.

4 A Sufficient Condition for URMT Tolerating $\mathcal{A} \subseteq \mathbb{A}$ with $|\bar{\mathcal{A}}| = 2$

We now give a sufficiency condition for the existence of URMT in $\mathcal{N}$ tolerating $\mathcal{A} \subseteq \mathbb{A}$ with $|\bar{\mathcal{A}}| = 2$.

Theorem 3 *Let $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ be a digraph under the influence of $\mathcal{A}$ with maximal basis $\overline{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$. Suppose $\mathcal{N}$ is such that for each $\alpha \in \{1, 2\}$, there exists a strong path (not necessarily distinct) p_α from $\mathbf{S}$ to $\mathbf{R}$, avoiding nodes from $(B_\alpha \cup F_\alpha)$. Furthermore, there exists a strong path q (not necessarily distinct from p_α's) from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$ which avoids nodes from $(B_1 \cup B_2 \cup (F_1 \cap F_2))$. Then there exists an URMT protocol tolerating $\mathcal{A}$.*

PROOF: According to the conditions of the theorem, there exists three strong paths p_1, p_2 and q (not necessarily distinct) from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$, such that p_1 avoids nodes from $(B_1 \cup F_1)$, p_2 avoid nodes from $(B_2 \cup F_2)$ and P avoid nodes from $(B_1 \cup B_2 \cup (F_1 \cap F_2))$. To reliably transmit a message m, $\mathbf{S}$ sends m along p_1, p_2 and q. Each intermediate node u along these paths forwards the message that it received to the corresponding neighbor. If nothing is received by the time something should have been received (since the network is synchronous, strict time-out conditions are feasible) then it forwards a new message namely "Null-from-u" to its neighbor. $\mathbf{R}$ now recovers m as follows: If $\mathbf{R}$ receives a valid message x along path q then $x = m$ because q is free from both B_1 and B_2. If a "Null-from-u" message is received along q, then if u's predecessor node in q belongs to F_1, then $\mathbf{R}$ outputs the message that is (guaranteed to be) received along path p_1. Else if u's predecessor node in q belongs to F_2, then $\mathbf{R}$ outputs the message that is (guaranteed to be) received along path p_2. However, if nothing is received along path q and if the $\mathbf{R}$'s predecessor in q belongs to F_1, then $\mathbf{R}$ outputs the message that is (guaranteed to be) received along path p_1, else $\mathbf{R}$ outputs the message that is (guaranteed to be) received along path p_2. □

Definition 8 *We call the USMT protocol given in Theorem 3 as protocol* Π.

4.1 Relaxing the Sufficiency Condition of Theorem 3

In the previous section, we have seen that if the paths p_1, p_2 and q are present in a network $\mathcal{N}$, then URMT is possible over $\mathcal{N}$. Now the question is whether the physical presence of the paths are necessary in $\mathcal{N}$? Well, the presence of p_1 and p_2 is obviously necessary for URMT in $\mathcal{N}$. Otherwise, the adversary can strategize to block all the strong paths from $\mathbf{S}$ to $\mathbf{R}$, thus preventing any kind of communication from $\mathbf{S}$ to $\mathbf{R}$. However, the physical presence of q in $\mathcal{N}$ is not necessary for URMT from $\mathbf{S}$ to $\mathbf{R}$! Here, we show that even in the absence of q, one can design URMT over $\mathcal{N}$ tolerating $\bar{\mathcal{A}}$, provided the effect of q can be simulated over $\mathcal{N}$. This is possible provided $\mathcal{N}$ satisfies certain conditions with respect to $\bar{\mathcal{A}}$.

Example 1: Consider the network $\mathcal{N}$ shown in Fig. 3, along with the adversary structure $\bar{\mathcal{A}}$. In $\mathcal{N}$,

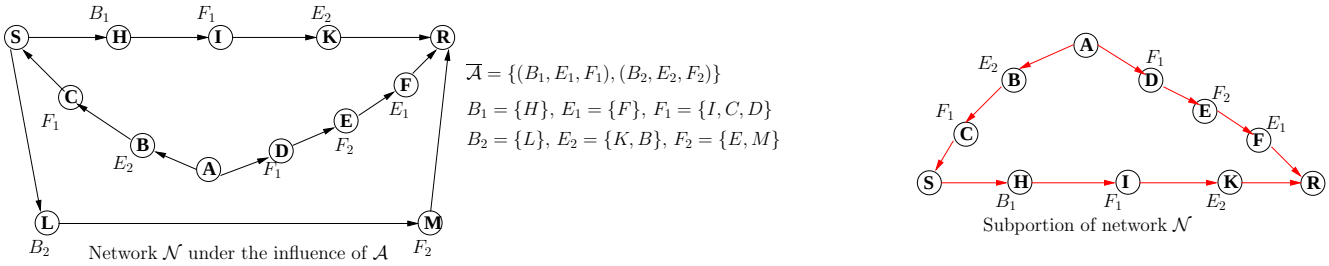


Figure 3: A network for Example 1

path $p_1 = (\mathbf{S}, L, M, \mathbf{R})$ is free from the nodes in $(B_1 \cup F_1)$, and path $p_2 = (\mathbf{S}, H, I, K, \mathbf{R})$ is free from the nodes in $(B_2 \cup F_2)$. However, there does not exist any strong path q which is free from the nodes in $(B_1 \cup B_2 \cup (F_1 \cap F_2))$. So $\mathcal{N}$ does not completely satisfy all the conditions of Theorem 3 with respect to the $\bar{\mathcal{A}}$. However, the effect of q can be simulated in $\mathcal{N}$.

Consider the sub-portion of $\mathcal{N}$ with strong path $(\mathbf{S}, H, I, K, \mathbf{R})$ and semi-strong path $(\mathbf{S}, C, B, A, D, E, F, \mathbf{R})$ (with head A), as shown in the second picture (drawn in red color) in Fig 3. Now consider the following sub-protocol called Π_1^{sim} executed over this sub-portion to send a value $s \in \mathbb{F}$ from $\mathbf{S}$ to $\mathbf{R}$: First, A sends three random secret keys $K_1, K_2, K_3 \in \mathbb{F}$ to $\mathbf{S}$ via the strong path $(A, B, C, \mathbf{S})$. If A does not receive all the keys, he uses three random keys of his own choice instead. A then sends $(x, y) = auth(s, K_1, K_2, K_3)$ along strong path $(\mathbf{S}, H, I, K, \mathbf{R})$ to $\mathbf{R}$. Now, A sends the same three keys (namely K_1, K_2 and K_3) to $\mathbf{R}$ along the strong path $(A, D, E, F, \mathbf{R})$. *Note that A sends the keys to $\mathbf{R}$ only after $\mathbf{S}$ has sent the authenticated message (namely (x_1, y_1)) along the strong path to $\mathbf{R}$. This can be done because the system is synchronous and the protocol is executed in rounds.* If $\mathbf{R}$ does not receive the keys from A, then $\mathbf{R}$ *knows* the identity of the set in $\overline{\mathcal{A}}$ that is corrupt. This can be achieved because if any node does not receive data from it's predecessor node, say j, then it generates a message saying *"message not received from node j"* and pass on to the next node in the path. Since the strong path from A to $\mathbf{R}$ contains nodes from F_1^* and F_2^*, $\mathbf{R}$ will always identify the corrupted set in $\overline{\mathcal{A}}$ if it does not receive keys from A. Similarly, if $\mathbf{R}$ does not receive any value from $\mathbf{S}$ along the strong path $(\mathbf{S}, H, I, K, \mathbf{R})$, then $\mathbf{R}$ can easily conclude that the first set in $\overline{\mathcal{A}}$ is corrupted. However, if $\mathbf{R}$ did receive the keys along path $(A, D, E, F, \mathbf{R})$ and tuple (x', y') along $(\mathbf{S}, H, I, K, \mathbf{R})$, then $\mathbf{R}$ verifies if $y' \stackrel{?}{=} x'K_2 + K_3$. If yes, then $\mathbf{R}$ outputs the message $(x' - K_1)$; else, B concludes that first set in $\overline{\mathcal{A}}$ is corrupted.

If the second set (B_2, E_2, F_2) of $\overline{\mathcal{A}}$ is corrupted, then the adversary will know K_1, K_2, K_3 (when A sends them to $\mathbf{S}$). But there is no node from B_2 along the strong path from $\mathbf{S}$ to $\mathbf{R}$. So, the authenticated message will reach correctly to $\mathbf{R}$. Since there are no nodes from B_2 along the strong path from A to $\mathbf{R}$, $\mathbf{R}$ will either correctly receive the keys or it will not receive any key, depending upon whether the node E crashes or not. If it crashes, then $\mathbf{R}$ will not receive any key but will know that second set in $\overline{\mathcal{A}}$ is corrupted. On the other hand, if $\mathbf{R}$ receives the keys from A, then they are correct and so the verification step at $\mathbf{R}$'s end will succeed and $\mathbf{R}$ will correctly output s.

If the first set (B_1, E_1, F_1) of $\overline{\mathcal{A}}$ is corrupted, then adversary will also know the keys by passively listening node F. But in the protocol, A sends the keys to $\mathbf{R}$, only after the authenticated message reaches to $\mathbf{R}$ through the strong path $(\mathbf{S}, H, I, K, \mathbf{R})$. So the node H, which is B_1 type corrupted will not know the keys when the authenticated message passes through H. *Hence the delay done by A in sending the keys to $\mathbf{R}$ plays a very significant role in the sub-protocol.* In essence, the node from B_1

on path $(\mathbf{S}, H, I, K, \mathbf{R})$ can not change the authenticated message in a consistent manner without being detected by $\mathbf{R}$ with very high probability. Now similar to information checking protocol of Rabin [9], adversary can forge the authenticated message with probability $\frac{1}{|\mathbb{F}|}$, without knowing K_1, K_2, K_3. Once the authenticated tuple reaches $\mathbf{R}$, adversary will also know the keys which A sends to $\mathbf{R}$. So now from the authenticated tuple which passed through H and the keys which passed through the node F, adversary can compute m. But now he cannot change it, as $\mathbf{R}$ already have either recovered m (if the authenticated tuple is received correctly) or knows that the first set in $\overline{\mathcal{A}}$ is corrupted.

Thus, what the above sub protocol achieves is the following: *The adversary has full information about the transmitted value s and also has complete control over* $\mathbf{R}$*'s output which could be either a valid message or a null message with the knowledge of the identity of the set in* $\overline{\mathcal{A}}$ *which is actually corrupt. Moreover, if* $\mathbf{R}$ *receives a valid message, it is indeed the correct message with a very high probability.* This is identical to saying that $\mathbf{S}$, with a very high probability, sends a message to $\mathbf{R}$ through nodes that are in F_1, E_1, F_2 and E_2 respectively. Thus Π_1^{sim} has the effect of simulating a "virtual path" between $\mathbf{S}$ and $\mathbf{R}$ with very high probability. So $\mathcal{N}$ in Fig. 3 can be enhanced to network $\mathcal{N}_1$ under the influence of $\bar{\mathcal{A}}_1$ as shown in Fig. 4 where in $\mathcal{N}_1$, there exists a "virtual path" between $\mathbf{S}$ and $\mathbf{R}$, containing intermediate virtual nodes X_1, X_2, X_3 and X_4, where $X_1 \in F_1, X_2 \in E_1, X_3 \in F_2$ and $X_4 \in E_2$ respectively. Now note that $\mathcal{N}_1$ satisfies the conditions of Theorem 3 with respect to $\bar{\mathcal{A}}_1$, where the virtual path $(\mathbf{S}, X_1, X_2, X_3, X_4, \mathbf{R})$ serves as path q. So the URMT protocol Π (of Theorem 3) can be executed over $\mathcal{N}_1$ tolerating $\bar{\mathcal{A}}$. But we want to design an URMT protocol over $\mathcal{N}$ which is the given physical graph. So we have to simulate the URMT protocol Π executed over $\mathcal{N}_1$ tolerating $\bar{\mathcal{A}}$, into an URMT protocol over $\mathcal{N}$ tolerating $\bar{\mathcal{A}}_1$. Our next goal is to demonstrate that simulation.

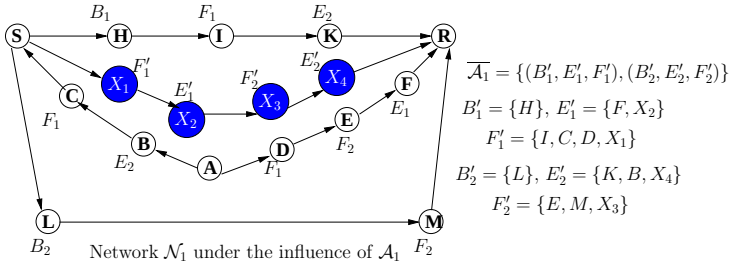


Figure 4: Network $\mathcal{N}$ updated to $\mathcal{N}_1$

Any value which is sent over p_1 or p_2 in protocol Π over $\mathcal{N}_1$ can be also sent over the same paths in $\mathcal{N}$ (as these paths are physically present in $\mathcal{N}$). Similarly, any value which is sent over the virtual path $(\mathbf{S}, X_1, X_2, X_3, X_4, \mathbf{R})$ in protocol Π over $\mathcal{N}_1$ can be also sent in $\mathcal{N}$ by using the sub-protocol Π_1^{sim}. [2] Thus all the steps of Π over $\mathcal{N}_1$ can be simulated over $\mathcal{N}$ also. If the error probability of sub-protocol Π_1^{sim} is δ' (which is at most $\frac{1}{|\mathbb{F}|}$), then the error probability of the protocol Π simulated over $\mathcal{N}$ is at most $n\delta'$, where n is the number of times sub-protocol Π_1^{sim} is executed. So we can make the error probability of resultant URMT protocol over $\mathcal{N}$ to be at most δ, by appropriately selecting $|\mathbb{F}|$ so that $n\delta' = \delta$.

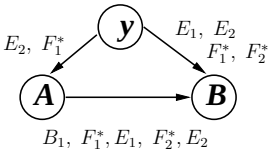
Summary of the example: In **Example 1**, we have seen a network, which do not satisfy the conditions of Theorem 3, but still protocol Π could be simulated on them with very high probability. In **Example 1**, we demonstrated a graph which contains a "special structures" (which satisfied some "special properties" with respect to $\bar{\mathcal{A}}$). This structure lead to the simulation of a special type of "virtual path" in the original network. Also, though not demonstrated, the "virtual path(s)" could be added *recursively*. Finally, the enhanced graph, with virtual path added, satisfies conditions of Theorem 3 and hence we could simulate Π on enhanced graph. But Π can be run on the original graph with the help of sub-protocols like Π_1^{sim}. So the idea is that starting from a physical graph (where all the edges and nodes are physical), we find the special structures (recursively) and keep on enhancing the graph (step by step through some intermediate graphs) until no more special structure is present on the (enhanced) graph. The final enhanced graph is named as **URMT-BEF-Closure-Digraph** of the original graph. If **URMT-BEF-Closure-Digraph** satisfies conditions of Theorem 3, then URMT protocol Π exists on the Closure graph. The protocol Π can be run on the physical (original) graph using the sub-protocols that simulate the respective virtual paths present in **URMT-BEF-Closure-Digraph**. In next section, we explore all possibilities of special structures and define **URMT-BEF-Closure-Digraph** formally.

[2] Note that each time an independent random triplet of keys are used to execute the sub-protocol Π_1^{sim}.

5 Definition of URMT-BEF-Closure-Digraph

Definition 9 (URMT-BEF-Closure-Digraph) *Let $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ be the network (directed graph) influenced by a non-threshold adversary characterized by the adversary structure $\mathcal{A}$ with a maximal basis of exactly two elements, say $\overline{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$. We inductively define a sequence of networks $\mathcal{N}_1, \mathcal{N}_2 \ldots$ where the set of vertices, denoted by $\mathbb{P}_i$, of the network $\mathcal{N}_i$ is defined as $\mathbb{P}_i = \mathbb{P} \cup \mathbb{V}_i$ with $\mathbb{V}_1 = \emptyset$ and the set of edges, say $\mathbb{E}_i$, of the network $\mathcal{N}_i$ is defined as $\mathbb{E}_i = \mathbb{E} \cup A_i$ with $A_1 = \emptyset$. The set V_i denotes the set of virtual nodes in $\mathcal{N}_i$, while A_i denotes the set of virtual edges in $\mathcal{N}_i$. We also define a corresponding sequence of adversary structures with maximal basis of two elements each, viz., $\mathcal{A}_1, \mathcal{A}_2, \ldots$, where $\mathcal{A}_1 = \mathcal{A}$. The details are as follows:*

The network $\mathcal{N}_i, i \geq 2$ can be constructed from the network $\mathcal{N}_{i-1}$ in four *different ways by applying one of the constructions from Table 1. In the table, a typical entry like*

$\#n$	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$ *where* $X_1 \in F_1, X_2 \in F_2$, $X_3 \in E_1$, $X_4 \in E_2$	• *Head* $\to A$: *avoids nodes from* $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$ **with condition** $\mathcal{Q}_1$ • *Head* $\to B$ *avoids nodes from* $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ • $A \to B$ *avoids nodes from* $((B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with condition** $\mathcal{Q}_2$	

means the following:
"In the n^{th} way of construction, we could potentially add an virtual path with four new virtual nodes X_1, X_2, X_3 and X_4 and five new virtual edges to $\mathcal{N}_{i-1}$ to obtain $\mathcal{N}_i$. Specifically, we add directed edges $(A, X_1), (X_1, X_2), (X_2, X_3), (X_3, X_4)$ and (X_4, B) if and only if the digraph $\mathcal{N}_{i-1} = (\mathbb{P}_{i-1}, \mathbb{E}_{i-1})$ is such that there exists two physical nodes A, B in $\mathcal{N}_{i-1}$, such that for the two elements (B_1, E_1, F_1) and (B_2, E_2, F_2) in $\overline{\mathcal{A}}_{i-1}$, both the following (1 and 2) are true:

1. *there does not exist four nodes $w_1 \in (\mathbb{V}_{i-1} \cap F_1), w_2 \in (\mathbb{V}_{i-1} \cap F_2), w_3 \in (\mathbb{V}_{i-1} \cap E_1)$ and $w_4 \in (\mathbb{V}_{i-1} \cap E_2)$ such that the edges $(A, w_1), (w_1, w_2), (w_2, w_3), (w_3, w_4)$ and (w_4, B) belong to $\mathbb{E}_{i-1}$. This means n^{th} construction has not been already used for nodes A and B. This is interpreted by the second column of the entry.*

2. *Both the following (a and b) hold:*

   (a) *there exists a semi-strong path, say q with head y from A to B in $\mathcal{N}_{i-1}$, such that the strong path from y to A avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$ and satisfies condition $\mathcal{Q}_1$ (possibly null). Similarly, the strong path from y to B avoids nodes from $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$. This is interpreted by the first two bulleted items in the third column of the entry.*

   (b) *there exists a strong path, say p from A to B in $\mathcal{N}_{i-1}$, such that p avoids nodes from $((B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$. The path p satisfies the condition $\mathcal{Q}_2$ (possibly null). This is the interpretation of the third bulleted item in the third column of the entry. Further in addition to $\mathcal{Q}_2$, the following condition must always be satisfied by p: for each $i \in \{1, 2\}$, every occurrence of a node from $(B_i \cup F_i) \setminus \{A, B\}$ (if any) in p is after the last occurrence of a node from $B_{\bar{i}} \setminus \{A, B\}$ (if any), where if $i = 1$ ($i = 2$), then $\bar{i} = 2$ ($\bar{i} = 1$). Though not explicitly specified in the entry, the last condition should be always satisfied by the strong path(s) from A to B in all the constructions.*

If one of the above two conditions (1 and 2) fails, we continue to work with $\mathcal{N}_{i-1}$ influenced by $\mathcal{A}_{i-1}$. However, if both of them are true, then we let $\mathbb{V}_i = \mathbb{V}_{i-1} \cup \{X_1, X_2, X_3, X_4\}$ which implies that $\mathbb{P}_i = \mathbb{P}_{i-1} \cup \{X_1, X_2, X_3, X_4\}$; and we let $A_i = A_{i-1} \cup \{(A, X_1), (X_1, X_2), (X_2, X_3), (X_3, X_4), (X_4, B)\}$ which implies $\mathbb{E}_i = \mathbb{E}_{i-1} \cup \{(A, X_1), (X_1, X_2), (X_2, X_3), (X_3, X_4), (X_4, B)\}$; finally we let the new nodes X_1, X_2, X_3 and X_4 to be added to F_1, F_2, E_1 and E_2 respectively. That is, if $\overline{\mathcal{A}}_{i-1} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$, then we let $\overline{\mathcal{A}}_i = \{(B_1, E_1 \cup \{X_3\}, F_1 \cup \{X_1\}), (B_2, E_2 \cup \{X_4\}, F_2 \cup \{X_2\})\}$." The figure in the fourth column of the entry denotes the complementary view of the conditions specified in the third column of the entry. The labels along the edges of the figure denote the the set of allowable adversarial nodes along the semi-strong path and strong path(s) between A and B. It is obvious, that honest nodes can be always present along these paths. For example, in the figure, we have put sets E_2 and F_1^* along the edge $y \to A$ which means that the nodes along the strong path from y to A can be completely honest (denoted by $\mathbf{H}$) or may contain nodes from sets E_2 and F_1^*, where $F_i^* = F_i \setminus (F_1 \cap F_2)$, $i \in \{1, 2\}$.

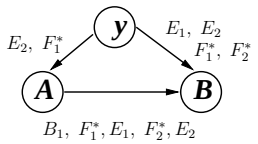
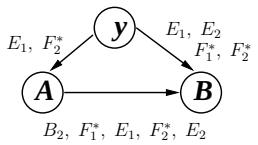
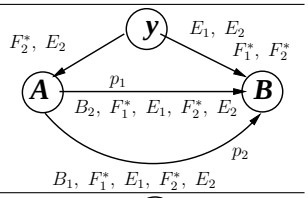
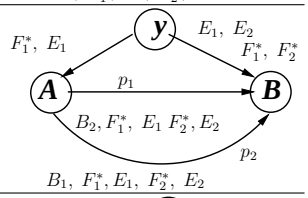
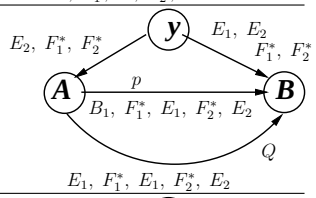
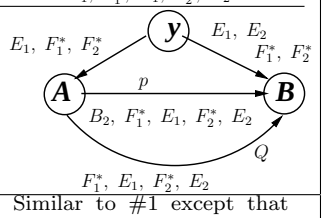
No.	Temporary Link	Conditions & Figure	
#1	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in F_1$, $X_2 \in E_1$, $X_3 \in F_2$, $X_4 \in E_2$	1. $y \to A$: $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: Path p: $((B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.**	
		1. $y \to A$: $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: Path p: $((B_1 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.**	
		1. $y \to A$: $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$ **with the last node from F_2^* before the first node from E_2.** 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: for each $i \in \{1, 2\}$, Path p_i: $(B_i \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.**	
		1. $y \to A$: $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1.** 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: for each $i \in \{1, 2\}$, Path p_i: $(B_i \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.**	
		1. $y \to A$: $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_1) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$, Path p: $(B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $A \to B$, Path Q: $(B_1 \cup B_2 \cup (F_1 \cap F_2)$	
		1. $y \to A$: $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_2) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$, Path p: $(B_1 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $A \to B$, Path Q: $(B_1 \cup B_2 \cup (F_1 \cap F_2)$	
#2	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in E_1$, $X_2 \in F_1$, $X_3 \in F_2$, $X_4 \in E_2$	Similar to the construction #1 except that the condition "with the last node from F_1^* before the first node from E_1" is removed from the strong path(s) from A to B in all the six cases	Similar to #1 except that first restriction on the ordering of vertices in the strong path(s) from A to B is relaxed
#3	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in F_1$, $X_2 \in E_1$, $X_3 \in E_2$, $X_4 \in F_2$	Similar to the construction #1 except that the condition "with the last node from F_2^* before the first node from E_2" is removed from the strong path(s) from A to B in all the six cases	Similar to #1 except that second restriction on the ordering of vertices in the strong path(s) from A to B is relaxed
#4	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in F_1$, $X_2 \in F_2$, $X_3 \in E_1$, $X_4 \in E_2$	Similar to the construction #1 except that both the conditions "with the last node from F_1^* before the first node from E_1" and "with the last node from F_2^* before the first node from E_2" are removed from the strong path(s) from A to B in all the six cases	Similar to #1 except that both restrictions on the ordering of vertices in the strong path(s) from A to B are relaxed

Table 1: The various constructions (#1 to #4) to augment $\mathcal{N}_{i-1}$ to $\mathcal{N}_i$. In the figures, y denotes the head of semi-strong path between A and B and the labels along the edges, represents the permissible category of adversary sets in permitted order for the construction. $F_i^* = F_i \setminus (F_1 \cap F_2)$ for $i \in \{1, 2\}$. Further in addition to the conditions specified in each construction, the strong path(s) from A to B must satisfy the following: for each $i \in \{1, 2\}$, every occurrence of a node from $(B_i \cup F_i) \setminus \{A, B\}$ is after the last occurrence of a node from $B_{\bar{i}} \setminus \{A, B\}$, where if $i = 1$ ($i = 2$), then $\bar{i} = 2$ ($\bar{i} = 1$)

Remark 2 *: A pair of vertices (A, B) may permit at most twenty-four augmentations, corresponding to one of the constructions from Table 1. When no augmentation is possible with respect to any pair of vertices, we stop the process. Thus, starting from $\mathcal{N}_1$, if we build a sequence of distinct networks $\mathcal{N}_1, \mathcal{N}_2, \cdots, \mathcal{N}_\nu$ through the augmenting process, we observe that $\nu \leq 24\binom{n}{2}$, where $n = |\mathbb{P}|$ denotes the set of nodes in $\mathcal{N}$. Also, we may consider the pairs of vertices in any order and augmentation may also*

be done in any order for a given pair of vertices. The URMT-BEF-closure-digraph of $\mathcal{N}$, denoted by $\mathcal{N}^*_{URMT_{BEF}}$ is defined as $\mathcal{N}^*_{URMT_{BEF}} = \mathcal{N}_\nu$. The corresponding adversary structure is $\mathcal{A}^* = \mathcal{A}_\nu$, where $|\bar{\mathcal{A}}^*| = 2$.

An illustration of constructing URMT-BEF-Closure-Digraph is given in **APPENDIX B**. We now briefly and informally mention few important properties of the constructions.

Property 1 (Principle Behind the Constructions) *In general, if $\mathcal{N}_{i-1}$ is augmented to $\mathcal{N}_i$ by applying some construction to A, B in $\mathcal{N}_{i-1}$ and if some value s is sent over the resultant virtual path from A to B in $\mathcal{N}_i$, then there always exist a sub-protocol Π^{sim} (as demonstrated in* **Example 1***), which when executed over $\mathcal{N}_{i-1}$ has one of the following outcomes: (a) Π^{sim} correctly sends s from A to B over $\mathcal{N}_{i-1}$ with negligible error probability, as demonstrated in* **Example 1***; (b) Π^{sim} may fail to send s, in which case it facilitates B to correctly know the exact identity of the corrupted set, as demonstrated in* **Example 1***. The basic format of the sub-protocol $\Pi^{simulate}$ will be more or less same for all the constructions (as shown in* **Example 1***). We do not provide the Π^{sim} protocol for every construction given in Table 1 due to space constraint.*

Lemma 1 *$\mathcal{N}^*_{URMT_{BEF}}$ has finite number of nodes and is unique (up to isomorphism).*

Proof: The finiteness property follows from the Remark 2 provided in the Definition 9. The proof of the uniqueness property is similar to the proof of Lemma 2 in [11] and hence is omitted[3]. □

Property 2 (Property of $\mathcal{A}^*$) *If $\bar{\mathcal{A}} = \bar{\mathcal{A}}_1 = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$ and $\bar{\mathcal{A}}^* = \{(B'_1, E'_1, F'_1), (B'_2, E'_2, F'_2)\}$, then we have $B'_1 = B_1, B'_2 = B_2$, $(F'_1 \cap F'_2) = (F_1 \cap F_2)$ and $(E'_1 \cap E'_2) = (E_1 \cap E_2)$. This is because the B_i's are never changed and no new virtual node is simultaneously added to both the fail-stop sets or both the passive sets at any stage in any of the constructions. Also note that each virtual node in $\mathcal{N}^*_{URMT_{BEF}}$ has a unique in-neighbor and out-neighbor.*

6 True Characterization of URMT Tolerating $\mathcal{A}$ with $|\bar{\mathcal{A}}| = 2$

We now give first ever true characterization of URMT in an arbitrary digraph $\mathcal{N}$ tolerating an adversary structure $\mathcal{A}$ with $|\bar{\mathcal{A}}| = 2$, in terms of $\mathcal{N}^*_{URMT_{BEF}}$. This along with Theorem 2, completely characterizes URMT in $\mathcal{N}$ tolerating any arbitrary adversary structure $\mathbb{A}$.

Theorem 4 *Let $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ be a directed graph, where $\mathbf{S}, \mathbf{R} \in \mathbb{P}$. Let $\mathcal{N}$ be under the influence of a non-threshold adversary $\mathcal{A}$ with maximal basis $\bar{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$. Furthermore, let $\mathcal{N}^*_{URMT_{BEF}} = (\mathbb{P}^*, \mathbb{E}^*)$ denotes the URMT-BEF-closure-digraph of network $\mathcal{N}$ with respect to $\mathcal{A}$. Moreover, let $\mathcal{N}^*_{URMT_{BEF}}$ be under the control of $\mathcal{A}^*$ where $\mathcal{A}^*$ is the adversary closure of $\mathcal{A}$ with maximal basis $\bar{\mathcal{A}}^* = \{(B'_1, E'_1, F'_1), (B'_2, E'_2, F'_2)\}$. Then URMT between $\mathbf{S}$ and $\mathbf{R}$ is possible in $\mathcal{N}$ tolerating $\mathcal{A}$ iff (a) for each $\alpha \in \{1, 2\}$, there exists a strong path (not necessarily distinct) p_α from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$ avoiding nodes from $(B_\alpha \cup F_\alpha)$ and (b) there exists a strong path P (not necessarily distinct from p_α's) from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}^*_{URMT_{BEF}}$, avoiding nodes from $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2))$.*

Proof: Sufficiency: Suppose the conditions of the theorem are true. Now closely looking at the conditions, we observe that they are almost same as the sufficiency conditions in Theorem 3. Now in order to prove the sufficiency of the Theorem 4, we begin with a definition.

Definition 10 ($URMT_{forward}$) *An URMT protocol over digraph $\mathcal{N}_i = (\mathbb{P} \cup \mathbb{V}_i, \mathbb{E} \cup \mathbb{A}_i)$ is called an $URMT_{forward}$ protocol, if in the protocol, the virtual nodes (nodes in $\mathbb{V}_i$) are capable of only receiving and forwarding messages and do no other computation; i.e., they do not use any internal random coins.*

In order to prove the sufficiency of the Theorem 4, we first show that if the conditions of Theorem 4 are satisfied, then we can design an $URMT_{forward}$ protocol over $\mathcal{N}^*_{URMT_{BEF}}$ tolerating $\bar{\mathcal{A}}^*$ (Lemma 2). We then show that if there exists an $URMT_{forward}$ protocol over $\mathcal{N}_i$ for $i > 1$ tolerating $\bar{\mathcal{A}}_i$, then there exists

[3]In [11], the authors have given the construction of closure graph by considering only Byzantine adversary. The constructions given here can be viewed as non-trivial generalization of the constructions given in [11]

an $URMT_{forward}$ protocol over $\mathcal{N}_{i-1}$ tolerating $\bar{\mathcal{A}}_{i-1}$ (Lemma 3). Now any $URMT_{forward}$ protocol over the original graph $\mathcal{N} = \mathcal{N}_1$ is actually an URMT protocol over $\mathcal{N}$. This is because there are no virtual nodes in $\mathcal{N}$; i.e., $\mathbb{V}_1 = \emptyset$. Since $\mathcal{N}^*_{URMT_{BEF}}$ is finite and unique (see Lemma 1), sufficiency of Theorem 4 follows from Lemma 2 and Lemma 3.

Lemma 2 *If the conditions of Theorem 4 are satisfied, then there exists an $URMT_{forward}$ protocol from $\mathbf{S}$ to $\mathbf{R}$ in the network $\mathcal{N}^*_{URMT_{BEF}}$ tolerating the adversary structure $\bar{\mathcal{A}}^*$.*

PROOF: It is easy to see that if the conditions of Theorem 4 are satisfied in $\mathcal{N}^*_{URMT_{BEF}}$ with respect to $\bar{\mathcal{A}}^*$, then protocol Π (of Theorem 3) can be executed over $\mathcal{N}^*_{URMT_{BEF}}$. Let us call the protocol as Π^*. It is easy to see that Π^* is an $URMT_{forward}$ protocol in $\mathcal{N}^*_{URMT_{BEF}}$ tolerating $\bar{\mathcal{A}}^*$ because in Π^*, the virtual nodes only receive and forward messages and do no other computation. □

Lemma 3 *For any $i > 1$, there exists an $URMT_{forward}$ protocol from $\mathbf{S}$ to $\mathbf{R}$ in the graph $\mathcal{N}_i$ tolerating the adversary structure $\bar{\mathcal{A}}_i$ if and only if there exists an $URMT_{forward}$ protocol from $\mathbf{S}$ to $\mathbf{R}$ in the network $\mathcal{N}_{i-1}$ tolerating the adversary structure $\bar{\mathcal{A}}_{i-1}$.*

PROOF: *If part:* This is the easy part. In fact, it is fairly obvious since any $URMT_{forward}$ protocol over $\mathcal{N}_{i-1}$ can be directly run over $\mathcal{N}_i$ *without* using the newly added virtual nodes at all! This is guaranteed to work because the adversary structure $\bar{\mathcal{A}}_i$ differs from $\bar{\mathcal{A}}_{i-1}$ only with respect to the virtual nodes that are newly added.

Only-if Part:(SKETCH) Let Π_i be an $URMT_{forward}$ protocol over $\mathcal{N}_i$ tolerating $\bar{\mathcal{A}}_i$. Using Π_i, we now design an $URMT_{forward}$ protocol Π_{i-1} over $\mathcal{N}_{i-1}$ tolerating $\bar{\mathcal{A}}_{i-1}$. Let $\mathcal{I}$ be an instruction in the protocol Π_i involving some nodes from $\mathcal{N}_i$. If all these nodes are also present in $\mathcal{N}_{i-1}$, then $\mathcal{I}$ can also be executed over $\mathcal{N}_{i-1}$. Hence $\mathcal{I}$ will be present in Π_{i-1}. On the other hand, suppose $\mathcal{I}$ is of the form "send M along (A, X_1) who then forwards it to X_2, who in turn forwards it to X_3, who in turn forwards it to X_4, who finally forwards it to B", such that A, B are physical nodes in $\mathcal{N}_i$ (and hence in $\mathcal{N}_{i-1}$) and X_1, X_2, X_3, X_4 are virtual nodes present in $\mathcal{N}_i$ but not in $\mathcal{N}_{i-1}$. In this case, $\mathcal{I}$ cannot be executed over $\mathcal{N}_{i-1}$ directly. But since X_1, X_2, X_3 and X_4 are virtual nodes present in $\mathcal{N}_i$ but not in $\mathcal{N}_{i-1}$, it implies that these virtual nodes would have been added to $\mathcal{N}_{i-1}$ by applying one of the constructions, say $\mathcal{C}$ Table 1, to the nodes A and B. However, as pointed out in Property 1, for construction $\mathcal{C}$ there is a sub-protocol $\Pi^{sim}_{\mathcal{C}}$ which can simulate the effect of the virtual path/edge (added by $\mathcal{C}$) over the graph $\mathcal{N}_{i-1}$ on which $\mathcal{C}$ is applied. An example of sub-protocol $\Pi^{sim}_{\mathcal{C}}$ protocol is provided in **Example 1** of previous section. Thus, we can replace the instruction $\mathcal{I}$, with sub-protocol $\Pi^{sim}_{\mathcal{C}}$ in Π_{i-1}. In this way, from Π_i, we get Π_{i-1}. This completes the proof of the lemma. □

Remark 3 *If the sub-protocol that we use to replace instruction $\mathcal{I}$ in $\mathcal{N}_i$ is incorrect with error probability δ_{sub} and the sub-protocol is invoked N times, then the resultant $URMT_{forward}$ protocol is incorrect with a probability up to $N\delta_{sub}$. Since, δ_{sub} can be reduced exponentially by a linear blow-up in the communication complexity (as the number of bits required to represent a field element is $\log |\mathbb{F}|$), we may set $\delta_{sub} = \frac{\delta}{N}$ where δ is the tolerance limit of the $URMT_{forward}$ protocol and N is an upper bound on the number of sub-protocol invocations. This increases the overall communication complexity by a factor of $O(\log \frac{N}{\delta})$.*

Now the proof of sufficiency of the Theorem 4 follows from the Lemma 1, 2 and 3. We now proceed to prove the necessity part of the Theorem 4.

NECESSITY (SKETCH): The necessity of path p_α in $\mathcal{N}$ is obvious. Otherwise all the strong paths from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$ will contain nodes from $(B_\alpha \cup F_\alpha)$ and the adversary can choose to corrupt the α^{th} set from $\bar{\mathcal{A}}$ and block all the nodes from $(B_\alpha \cup F_\alpha)$, thus refuting any type of communication from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$. Finally the necessity of path P in $\mathcal{N}^*_{URMT_{BEF}}$ is proved by contradiction. Suppose there exists an URMT protocol Π^* in $\mathcal{N}^*_{URMT_{BEF}}$ (and hence in $\mathcal{N}$) tolerating $\bar{\mathcal{A}}^*$ even in the absence of path p_P in $\mathcal{N}^*_{URMT_{BEF}}$. Since P does not exist, it implies that each of the strong paths from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}^*_{URMT_{BEF}}$ contain nodes from $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2))$. We now divide the set of nodes (virtual + physical) in $\mathcal{N}^*_{URMT_{BEF}}$ as follows: let Y_1 be the set of all nodes that have a strong path to $\mathbf{R}$ in $\mathcal{N}^*_{PPSMT_{BEF}}$ that does not use any vertex from $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2))$. Furthermore, let $X_1 = \mathbb{P}^* \setminus (B'_1 \cup B'_2 \cup (F'_1 \cap F'_2) \cup Y_1)$. Clearly, $\mathbf{R} \in Y_1$ and $\mathbf{S} \in X_1$. Moreover, it is evident from the definition of Y_1 that there are no edges from any node in X_1

to any node in Y_1. However, there can be some reverse path(s) from the node(s) in Y_1 to the node(s) in X_1. The necessity of P is now proved in two parts:

1. We first show that if there are no reverse path(s) from the node(s) in Y_1 to the node(s) in X_1, then in the absence of P, there always exists an adversary strategy using which $\bar{\mathcal{A}}^*$ can violate the reliability property of Π^* (see Lemma 5 in **APPENDIX C**).

2. We next show that even if there is some reverse path, say p, from Y_1 to X_1, then also presence of p does not help in the possibility of URMT (in the absence of P), thereby maintaining the impossibility of URMT in $\mathcal{N}^*_{URMT_{BEF}}$ as projected by Lemma 5. This is tricky to prove. In order to prove this, we consider all possible allowable behavior of path p. We then show that corresponding to each different status of p, the strong path(s) from X_1 to Y_1 should definitely satisfy certain properties. If not, then we could augment $\mathcal{N}^*_{URMT_{BEF}}$ by applying at least one of the constructions, thus contradicting the fact that $\mathcal{N}^*_{URMT_{BEF}}$ is **URMT-BEF-Closure-Digraph**. Now once it is shown that corresponding to each status of p, the strong path(s) from X_1 to Y_1 exhibit certain properties, we prove that there always exists an adversary strategy which disallows p to help in the possibility of URMT at all.

So existence of P is necessary for possibility of Π^* on $\mathcal{N}^*_{URMT_{BEF}}$. This in turn implies the necessity of P in $\mathcal{N}^*_{URMT_{BEF}}$ for the possibility of URMT in $\mathcal{N}$. For the complete proof of the above two cases, see **APPENDIX C**. □

Theorem 4 is demonstrated with an example in **APPENDIX I**. We have thus characterized URMT in an arbitrary directed graph tolerating a non-threshold mixed adversary. As stated earlier this also provides the characterization for the possibility of URMT on $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$. Now what is left is the characterization of $URMT_{special}$ on $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$. For this, in the next section, we characterize $URMT_{special}$ on $\mathcal{D}_{under}$ tolerating a non-threshold mixed adversary $\mathbb{A}$.

7 Characterization of $URMT_{special}$ on $\mathcal{D}_{under}$

In this section, we characterize $URMT_{special}$ (see Definition 6) in the "underlying digraph" $\mathcal{D}_{under}$ (Definition 5) of an arbitrary directed hypergraph $\mathcal{D}$, tolerating a non-threshold mixed adversary $\mathbb{A}$. This characterization, along with Theorem 1 completely characterize URMT in $\mathcal{D}$ tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$. Now similar to Theorem 2, we can reduce the problem of characterizing $URMT_{special}$ in $\mathcal{D}_{under}$ tolerating $\mathbb{A}$, to the problem of characterizing $URMT_{special}$ in $\mathcal{D}_{under}$ tolerating all possible $\mathcal{A} \subseteq \mathbb{A}$ where $\overline{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$. Now the only reason why a $URMT$ protocol may exist in $\mathcal{D}_{under}$ tolerating $\mathcal{A}$ but a $URMT_{special}$ does not exist in $\mathcal{D}_{under}$ (tolerating $\mathcal{A}$) is that in $URMT_{special}$ protocol, the nodes in $\mathcal{V}$ in $\mathcal{D}_{under}$ are forced to toss coins (which according to Definition 6, they cannot do). Now from the proof of sufficiency of Theorem 4, the problem comes when in the protocol, a node y from $\mathcal{V}$ in $\mathcal{D}_{under}$ is acting as the head of semistrong path between two nodes A and B (in one of the constructions in Table 1) and is forced to send some random secret keys $K_1, K_2, K_3 \in \mathbb{F}$ to A and B (as done in the sub-protocol in **Example 1**). Since the virtual nodes from $\mathcal{V}$ cannot do any random computation in $URMT_{special}$ protocol, we have to modify the definition of URMT-BEF-Closure-Digraph to obtain $URMT_{special}$-BEF-Closure-Digraph of $\mathcal{D}_{under}$ under the influence of $\mathcal{A}$. We highlight only the modifications.

Definition 11 ($URMT_{special}$-BEF-Closure-Digraph) *Let $\mathcal{N} = \mathcal{D}_{under} = (\mathcal{P}', \mathcal{E}')$ be the "underlying digraph" of a directed hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$, where $\mathcal{P}' = (\mathcal{P} \cup \mathcal{V})$. Let $\mathcal{D}_{under}$ be under the influence of a non-threshold mixed adversary $\mathcal{A}$ with exactly two elements in its maximal basis $\overline{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$. We inductively define a sequence of directed networks $\mathcal{N}_1, \mathcal{N}_2 \ldots$ with $\mathcal{N}_1 = \mathcal{N}$, where the set of vertices, denoted by $\mathcal{P}'_i$, of the network $\mathcal{N}_i$ is defined as $\mathcal{P}'_i = \mathcal{P}' \cup \mathbb{V}_i$ with $\mathbb{V}_1 = \emptyset$ and the set of edges, say $\mathcal{E}'_i$, of the network $\mathcal{N}_i$ is defined as $\mathcal{E}'_i = \mathcal{E}' \cup A_i$ with $A_1 = \emptyset$. We also define a corresponding sequence of adversary structures with two elements each, viz., $\mathcal{A}_1, \mathcal{A}_2, \ldots$, where $\mathcal{A}_1 = \mathcal{A}$.*

The network $\mathcal{N}_i$ is augmented from $\mathcal{N}_{i-1}$ by applying different constructions from Table 1 (as done in Definition 9), with certain additional restrictions imposed. We mention these restrictions. Let A, B

be two nodes in $\mathcal{N}_{i-1}$, where both $A, B \in \mathcal{P}$. Thus A, B are physical nodes in $\mathcal{D}_{under}$. Let p be a strong path between A, B and q be a semi-strong path between A, B with head y. Now suppose that paths p and q satisfy the condition of one of the constructions in Table 1, say $\mathcal{C}$. Let according to $\mathcal{C}$, S_A and S_B denotes the set of adversarial nodes which should be absent along the strong path from y to A and B respectively. Now we can augment $\mathcal{N}_{i-1}$ by applying $\mathcal{C}$ to A and B, if one of the following (extra) conditions are satisfied:
1. **The head y of q is such that $y \in \mathcal{P}$ (y is a Physical node in $\mathcal{N}_{i-1}$):** *In this case, $\mathcal{C}$ is directly applied to A and B (as done in Definition 9).*
2. **The head y of q is such that $y \in \mathcal{V}$ (y is a Virtual node in $\mathcal{N}_{i-1}$):** *In this case we put some additional constraints as follows: Let $\vec{z} \in \mathcal{P}$ be the immediate out-neighbor of y on the path from y to B and $\vec{x} \in \mathcal{P}$ be the unique in-neighbor of y. Note that according to the definition of $\mathcal{D}_{under}$ (see Definition 5), if $y \in \mathcal{V}$, then it implies that the out-neighbor of y on the path from y to B ($\vec{z}$) and the in-neighbor of y ($\vec{x}$) are physical nodes in $\mathcal{D}_{under}$. Hence both $\vec{x}$ and $\vec{z}$ are physical nodes in $\mathcal{N}_{i-1}$. Since $y \in \mathcal{V}$ and is not allowed to toss random coins, it cannot perform any synchronisation; i.e., in any protocol, we cannot ask y to send some "secret" information to A first and send the same "secret" information to B, only after A has send some "authenticated" message to B (this principle is used in the sub-protocol in* **Example 1***). With these properties of $y, \vec{z}$ and $\vec{x}$, we now mention the additional constraints on $\vec{z}$ and $\vec{x}$:*

1. *Restriction on $\vec{z}$: $\vec{z} \notin (S_A \cap (E_1 \cup E_2)) \cup S_B$.*

   Remark: *Informally, the above restriction says that $\vec{z}$ can not be under the influence of a passive adversary set which is not allowed over the path from y to A in $\mathcal{C}$. This is so because in any protocol, y being a virtual node ($\in \mathcal{V}$), sends every information received from its in-neighbor $\vec{x}$, simultaneously to both $\vec{z}$ and the first node from y to A. So if $\mathcal{C}$ requires the path from y to A should not contain nodes from certain type of passive adversary set, then the same type of passive adversary set should not influence $\vec{z}$ too. With this restriction on $\vec{z}$, if any synchronisation is needed from y in the protocol (we have used such synchronisation from y when y is a physical node in the sub-protocol in* **Example 1***), can be taken care by $\vec{z}$. We will explain this more elaborately in the proof of Theorem 5.*

2. *Restriction on $\vec{x}$: (a) $\vec{x} \notin (F_1 \cap F_2)$ and (b) If $E_i \in S_A$, then $\vec{x} \notin E_i \cup B_i$, for $i \in \{1, 2\}$*

   Remark: *When the head y of path q is virtual node, all the computations supposed to be done by y in the protocol, is actually done by $\vec{x}$. The above restriction says that if path from y to A should devoid of E_i, then $\vec{x}$ can not belong to $E_i \cup B_i$. In addition, $\vec{x} \notin F_1 \cap F_2$. We prove the necessity and sufficiency of the restriction in sequel.*

If $\vec{z}$ and $\vec{x}$ follows the above restrictions, then we can augment $\mathcal{N}_{i-1}$ by applying $\mathcal{C}$ to A and B. Otherwise, $\mathcal{C}$ cannot be applied to A and B. Since $\mathcal{C}$ can be any of the 24 constructions from Table 1, we get 24 corresponding additional constructions when $y \in \mathcal{V}$. These additional constructions are given in Table 2 in **APPENDIX IV**.

Remark: A pair of vertices $(A, B) \in \mathcal{P}$ in $\mathcal{N}_{i-1}$ may permit at most 48 augmentations (24 according to Table 1 and 24 according to Table 2). When there is no augmentation possible with respect to any pair of vertices, we stop the process. Thus, starting from $\mathcal{N} = \mathcal{N}_1$, if we build a sequence of distinct networks $\mathcal{N}_1, \mathcal{N}_2, \cdots, \mathcal{N}_\nu$ through the augmenting process, we observe that $\nu \leq 48\binom{n}{2}$, where n is the number of physical nodes in $\mathcal{N}_1 = \mathcal{N}$. Also, we may consider the pairs of vertices in any order and augmentation may also be done in any order for a given pair of vertices. The $URMT_{special}$-BEF-Closure-Digraph of $\mathcal{N}$ denoted $\mathcal{N}^*_{URMTspl_{BEF}}$ is defined as $\mathcal{N}^*_{URMTspl_{BEF}} = \mathcal{N}_\nu$. The corresponding adversary structure $\mathcal{A}^*_{spl} = \mathcal{A}_\nu$. Note that Lemma 1 and Property 2 will be true for $\mathcal{N}^*_{URMTspl_{BEF}}$ also.

Theorem 5 *Let $\mathcal{N} = \mathcal{D}_{under} = (\mathcal{P}', \mathcal{E}')$ be the "underlying digraph" of an arbitrary directed hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$, where $\mathcal{P}' = \mathcal{P} \cup \mathcal{V}$ and $\mathcal{V}$ is the set of virtual nodes. Then $URMT_{special}$ is possible in $\mathcal{N}$ tolerating a non-threshold adversary $\mathbb{A}$ iff for every $\mathcal{A} \subseteq \mathbb{A}$ with $\overline{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$ both the following hold:*

1. *The network $\mathcal{N}$ is such that for each $\alpha \in \{1,2\}$ the deletion of nodes in $((B_\alpha \cup F_\alpha) \setminus \{\mathbf{S}, \mathbf{R}\})$ does not eliminate all the strong paths from $\mathbf{S}$ to $\mathbf{R}$.*

2. *The $URMT_{special}$-BEF-Closure-Digraph of network $\mathcal{N}$ with respect to the adversary structure $\mathcal{A}$, viz., $\mathcal{N}^*_{URMTspl_{BEF}} = (\mathbb{P}^*, \mathbb{E}^*)$ is such that, there exists a strong path from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}^*_{URMTspl_{BEF}}$, induced by the set of vertices $(\mathbb{P}^* \setminus (B'_1 \cup B'_2 \cup (F'_1 \cap F'_2))) \cup \{\mathbf{S}, \mathbf{R}\}$ where the adversary structure closure $\mathcal{A}^*_{spl} = \{(B'_1, E'_1, F'_1), (B'_2, E'_2, F'_2)\}$.*

PROOF: We will draw very heavily from the proof of Theorem 4, where we proved the necessary and sufficient condition for the existence of URMT — the conditions are analogous to the current theorem, the only difference is that extra restrictions (on $\vec{x}$ and $\vec{z}$) are imposed while forming closure graph. We now show that the extra restrictions are necessary and sufficient to convert URMT protocol into $URMT_{special}$ protocol. This completes the proof of current theorem, since the characterization of URMT is done in Theorem 4.

Suppose that one of the constructions in Definition 11 is that "there exists a semi-strong path q from A to B, with head y, such that the strong path from y to A does not contain nodes from the set S_A and the strong path from y to B does not contain nodes from the set S_B". We know from Theorem 4 that this construction is necessary and sufficient for the existence of URMT in $\mathcal{D}_{under}$, provided $y \in \mathcal{P}$; i.e., y is a non-virtual node in $\mathcal{D}_{under}$. Now let $y \in \mathcal{V}$ and $\vec{x} \in \mathcal{P}$ be the unique in-neighbor of y and $\vec{z} \in \mathcal{P}$ be the first node on the strong path between y and B. We now show that the extra restrictions imposed on $\vec{x}$ and $\vec{z}$ as per Definition 11 are necessary and sufficient for converting the URMT protocol over $\mathcal{D}_{under}$ into a $URMT_{special}$ protocol over $\mathcal{D}_{under}$, when $y \in \mathcal{V}$.

Necessity: We first note that in any $URMT_{special}$ protocol, if y is virtual node then all its random coin toss and computations are done by its unique in-neighbor $\vec{x}$. The first restriction on $\vec{x}$ is necessary because if $\vec{x} \in (F_1 \cap F_2)$ and if the adversary crashes $\vec{x}$, then neither A nor B will come to know the identity of the corrupted set. The second restriction on $\vec{x}$; i.e., if $E_i \in S_A, i \in \{1,2\}$, then $\vec{x} \notin (B_i \cup E_i)$, is necessary because of the following reason: We prove it by contradiction. If $y \in \mathcal{P}$, then from Theorem 4, the condition $E_i \in S_A$ is necessary for URMT. Now if $y \in \mathcal{V}$, then in the protocol, its random coins are tossed by $\vec{x}$. Hence it is necessary that $\vec{x} \notin (B_i \cup E_i)$. This is so because if $\vec{x} \in (B_i \cup E_i)$ and if the i^{th} set is corrupted, then the adversary will know the random coins tossed by $\vec{x}$ (on behalf of $\vec{y}$). This exactly simulates the scenario when $y \in \mathcal{P}$ and is corrupted by E_i. This implies that nodes from E_i are present in the strong path from y to A and hence $E_i \notin S_A$. This is a contradiction because $E_i \in S_A$.

The necessity of the restrictions on $\vec{z}$; i.e., $\vec{z} \notin (S_A \cap (E_1 \cup E_2)) \cup S_B$, can be argued in the following way: The restriction $\vec{z} \notin S_B$ is obvious from the fact that $\vec{z}$ is a node on the strong path from y to B. Now we argue that why $\vec{z} \notin (S_A \cap (E_1 \cup E_2))$. On the contrary, assume that $\vec{z} \in (S_A \cap (E_1 \cup E_2))$. This implies that $\vec{z}$ could be passively corrupted by a set $E_i, i \in \{1,2\}$, where $E_i \in S_A$. Now as mentioned earlier, if $y \in \mathcal{V}$, then all the local random coins (such as some local random values) are computed by $\vec{x}$ on behalf of $\vec{y}$. Moreover, $\vec{x}$ simultaneously delivers these random values to all the out-neighbors of y (by the property of hyperedge). So, both $\vec{z}$ and first node on the strong path from y to A receive these random values simultaneously from $\vec{x}$ via y. However, since $\vec{z} \in E_i$, if the i^{th} set is corrupted, then adversary will also know the random coin toss of $\vec{x}$ (done on behalf of y). This simulates as if $y \in E_i$, which implies that nodes from the set E_i are present along the strong path from y to A. This again implies that $E_i \notin S_A$, which is a contradiction.

Sufficiency: The URMT protocol designed in the proof of Theorem 4 can be easily converted into a $URMT_{special}$ protocol, if the heads of the semi-strong path considered in Definition 11 are all non-virtual. In case they are virtual, it is sufficient if the nodes $\vec{x}$ and $\vec{z}$ satisfy the restrictions imposed on them according to Definition 11. The conversion is as follows: For every random coin toss that should be done by a virtual node y in URMT protocol, is now done by $\vec{x}$ (on behalf of y) in $URMT_{special}$. $\vec{x}$ then simultaneously sends random coin tosses (values) to both $\vec{z}$ and the first node on the strong path from y to A (via y). Now any synchronisation which needs to be done by y (such as not sending the random values to B, till A receives them and sends an authenticated message to B, etc) in URMT protocol, will be taken care by $\vec{z}$ in $URMT_{special}$ protocol. Note that the restrictions imposed on $\vec{x}$ and $\vec{z}$ ensures that

$URMT_{special}$ protocol will retain all properties of URMT protocol. This is so because $\vec{x} \notin (F_1 \cap F_2)$. So if $\vec{x}$ crashes and fails to do random coin toss on behalf of y, then both A and B will come to know the identity of corrupted set. Similarly if $E_i \in S_A, i \in \{1,2\}$, then $\vec{x} \notin (B_i \cup E_i)$ and $\vec{z} \notin E_i$. In this case even if $\vec{i}^{th}$ set gets corrupt and listens the random coin toss done by $\vec{x}$, it will simulate as if $y \in E_{\vec{i}}$. However, this does not affect the correctness of the $URMT_{special}$ protocol since this adversarial behavior is countered in URMT protocol. This completes the sufficiency proof. □

Thus, we have characterized $URMT_{special}$ on $\mathcal{D}_{under}$ tolerating a non-threshold adversary $\mathbb{A}$. Now Theorem 5, along with Theorem 1, completely characterize URMT over directed hypergraph $\mathcal{D}$, tolerating $\mathcal{A}_{(t_b,t_f,t_p)}$. In **APPENDIX V**, we demonstrate the characterization of URMT on directed hypergraph (Theorem 5 + Theorem 1) on hypergraph $\mathcal{D}$ (shown in Fig. 2) tolerating two different adversary $\mathcal{A}_{(1,0,0)}$ ($t_b = 1$ and $t_p = t_f = 0$) and $\mathcal{A}_{(1,1,0)}$ ($t_b = t_p = 1$ and $t_f = 0$). We show that URMT is possible over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$ but impossible tolerating $\mathcal{A}_{(1,1,0)}$ (as claimed in subsection 1.3).

8 Conclusion

In this work, we focused on the problem of URMT over directed hypergraphs tolerating a threshold mixed adversary $\mathcal{A}_{(t_b,t_p,t_f)}$. We completely characterize URMT in directed hypergraph tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$. For that first we proved that URMT protocol on directed hypergraph tolerating $\mathcal{A}_{(t_b,t_p,t_f)}$ is possible iff a special kind URMT protocol $URMT_{special}$ is possible on the "underlying digraph" $\mathcal{D}_{under}$ tolerating a non-threshold mixed adversary. So, we first characterize the possibility of URMT in arbitrary digraphs tolerating *non-threshold mixed* adversary, then describe the modifications needed to adapt the characterization for $URMT_{special}$. Finally, we would like to remark that all the results presented in this paper are highly non-trivial and shows in-depth insight on this area and above all reveals many quintessential surprises.

References

[1] Amos Beimel and Matthew K. Franklin. Reliable communication over partially authenticated networks. In *WDAG*, pages 245–259, 1997.

[2] Y. Desmedt and Y. Wang. Perfectly secure message transmission revisited. In *Proceedings of Advances in Cryptology EUROCRYPT'02*, volume 2332 of *LNCS*, pages 502–517. Springer Verlag, 2002.

[3] Y. Desmedt and Y. Wang. Perfectly secure message transmission revisited. Cryptology ePrint Archive, Report 2002/128, 2002. url - http://eprint.iacr.org.

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

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

[6] M. Franklin and M. Yung. Secure hypergraphs: Privacy from partial broadcast. In *Proc. of 27th Ann. Symposium on Theory of Computing*, pages 36–44, 1995.

[7] M. V. N. A. Kumar, P. R. Goundan, K. Srinathan, and C. Pandu Rangan. On perfectly secure communication over arbitrary networks. In *Proc. of 21st PODC*, pages 193–202. ACM Press, 2002.

[8] Nancy Lynch. *Distributed Algorithms.* Morgan Kaufmann, San Mateo, CA, USA, 1996.

[9] T. Rabin and M. Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority. In *Proc. of 21st ACM STOC*, pages 73–85, 1989.

[10] B. Shankar, P. Gopal, K. Srinathan, and C. Pandu Rangan. Unconditional reliable message transmision in directed networks. In Proc. of SODA 2008.

[11] K. Srinathan and C. Pandu Rangan. Possibility and complexity of probabilistic reliable communication in directed networks. In *Proc. of 25th PODC*, pages 265–274. ACM Press, 2006.

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

APPENDIX A: Proof of Theorem 2

The only-if direction is obvious. For the if-direction, we show that if a URMT protocol exists while tolerating every monotone subset $\mathcal{A} \subseteq \mathbb{A}$ such that $|\overline{\mathcal{A}}| = 2$, then one can construct a URMT protocol that tolerates $\mathbb{A}$. Suppose that every monotone subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 2$, is tolerable. In order to show that every monotone subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 3$, is also tolerable, we argue as follows: for any subset $\mathcal{A} \subseteq \mathbb{A}$ with $|\overline{\mathcal{A}}| = 3$, there exist three subsets, each of size two, such that any element in $\overline{\mathcal{A}}$ belongs to exactly two of them. Specifically, we may choose to divide $\overline{\mathcal{A}} = \{x_1, x_2, x_3\}$ (where each x_i is an ordered triple (B_i, E_i, F_i)) into $\mathcal{A}_1 = \{x_1, x_2\}$, $\mathcal{A}_2 = \{x_2, x_3\}$ and $\mathcal{A}_3 = \{x_1, x_3\}$. Therefore, we may run three URMT protocols in parallel transmitting the same message, tolerating the adversary structures $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{A}_3$ respectively. $\mathbf{R}$ can output the majority among the individual outputs as the received message. Since any element from $\mathcal{A}$ belongs to at least two of the three $\mathcal{A}_i$'s, the above protocol is a URMT protocol tolerating $\mathcal{A}$ (since $\mathbf{R}$ gets the correct message in at least two of the three sub-protocols). Thus any subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 3$, is tolerable. Applying this procedure again, we find that any subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 4$, is tolerable — because any set of size four can be divided into three subsets of size three each, such that every element occurs in at least two of the subsets. In general, any $\mu > 3$ sized set can be divided into three subsets, each of size $\left\lceil \frac{2\mu}{3} \right\rceil$, such that every element of the set occurs in at least two of them. The rest follows from induction[4]. □

APPENDIX B: Illustration of Constructing URMT-BEF-Closure-Digraph $\mathcal{N}^*_{URMT_{BEF}}$

Consider a digraph $\mathcal{N}$, under the influence of an adversary structure $\mathcal{A} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$, as shown in the Fig. 5. We construct the URMT-BEF-Closure-Digraph of $\mathcal{N}$ by a sequence of application of constructions from Table 1. Initially let $\mathcal{N}_1 = \mathcal{N}$ and $\mathcal{A}_1 = \mathcal{A}$. Now consider nodes C and $\mathbf{R}$. The semi-strong path $(C, D, E, F, \mathbf{R})$ with head D and the strong path $(C, G, H, \mathbf{R})$ from C to $\mathbf{R}$ satisfies all the conditions required in Construction #1 subcase 2 (see the figure given in 2^{nd} row of Table 1). Hence in digraph $\mathcal{N}_1$, we apply Construction #1 subcase 2 between nodes C and $\mathbf{R}$ and add four new temporary vertices X_1, X_2, X_3 and X_4 and five new temporary edges namely, $(C, X_1), (X_1, X_2), (X_2, X_3), (X_3, X_4)$ and $(X_4, \mathbf{R})$. We denote this new digraph by $\mathcal{N}_2$. We also upgrade our adversary structure $\mathcal{A}_1$ such that $F_1 = F_1 \cup \{X_1\},\ F_2 = F_2 \cup \{X_3\}, E_1 = E_1 \cup \{X_2\}, E_2 = E_2 \cup \{X_4\}$. This augmented adversary structure is denoted by $\mathcal{A}_2$. Now consider the nodes $\mathbf{S}$ and $\mathbf{R}$ and the semi-strong path $(\mathbf{S}, A, B, C, X_1, X_2, X_3, X_4, \mathbf{R})$ with head C and strong path $(\mathbf{S}, I, J, K, \mathbf{R})$ between them in $\mathcal{N}_2$. The semi-strong path and the strong path between $\mathbf{S}$ and $\mathbf{R}$ satisfies all the conditions required in Construction #1 subcase 1 (see the figure given in 1^{st} row of Table 1). Therefore in $\mathcal{N}_2$, we apply Construction #1 subcase 1 between $\mathbf{S}$ and $\mathbf{R}$ and essentially add four new temporary vertices X_5, X_6, X_7 and X_8 and five new temporary edges namely, $(\mathbf{S}, X_5), (X_5, X_6), (X_6, X_7), (X_7, X_8)$ and $(X_8, \mathbf{R})$. We denote this new digraph by $\mathcal{N}_3$. We also upgrade our adversary structure $\mathcal{A}_2$ such that $F_1 = F_1 \cup \{X_5\},\ F_2 = F_2 \cup \{X_7\}, E_1 = E_1 \cup \{X_6\}, E_2 = E_2 \cup \{X_8\}$. This augmented adversary structure is denoted by $\mathcal{A}_3$. Now notice that in the digraph $\mathcal{N}_3$, there is no pair of *physical* nodes to which we can apply any *new* construction from Table 1. Hence, the augmenting process will stop here and the URMT-BEF-Closure-Digraph is defined as $\mathcal{N}^*_{URMT_{BEF}} = \mathcal{N}_3$ and corresponding adversary structure $\mathcal{A}^* = \mathcal{A}_3$. Let $\mathcal{A}^* = \{(B'_1, E'_1, F'_1), (B'_2, E'_2, F'_2)\}$.

Now we see that digraph $\mathcal{N}$, along with $\mathcal{A}$, satisfies the conditions specified in Theorem 3 and thus URMT is possible in $\mathcal{N}$ tolerating $\mathcal{A}$. The strong paths (from $\mathbf{S}$ to $\mathbf{R}$) that do not contain nodes from

[4]The protocol given here is based on induction and is used to only prove the sufficiency. It is not an efficient protocol

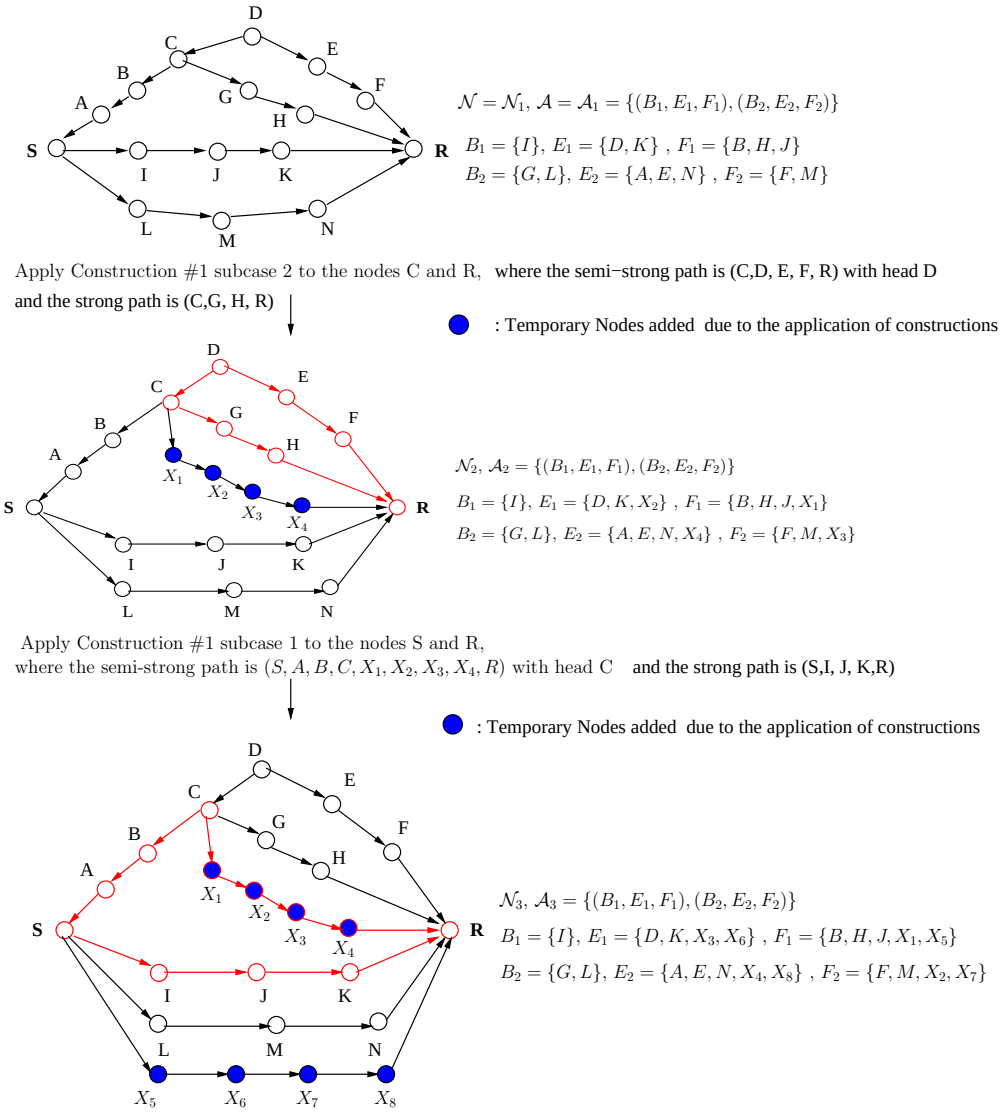


Figure 5: Illustration of constructing URMT-BEF-Closure-Digraph $\mathcal{N}^*_{URMT_{BEF}}$ of digraph $\mathcal{N}$

$(B_1 \cup E_1 \cup F_1)$ and $(B_2 \cup E_2 \cup F_2)$ are $(\mathbf{S}, L, M, , N, \mathbf{R})$ and $(\mathbf{S}, I, J, , K, \mathbf{R})$ respectively. In URMT-BEF-Closure-Digraph $\mathcal{N}^*_{URMT_{BEF}}$, the strong path (from $\mathbf{S}$ to $\mathbf{R}$) that is free from $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2)) = (B_1 \cup B_2 \cup (F_1 \cap F_2))$ is $(\mathbf{S}, X_5, X_6, X_7, X_8, \mathbf{R})$. Hence $URMT$ is possible in $\mathcal{N}$ tolerating $\mathcal{A}$.

APPENDIX C: Necessity Proof of Theorem 4

Necessity of path(s) p_1 and p_2 is already shown in Section 6. We now prove the necessity of path P. To prove the necessity of path P, we show that if there is no $URMT_{forward}$ protocol in $\mathcal{N}^*_{URMT_{BEF}}$, then there is no URMT protocol in $\mathcal{N}$ (Lemma 4). We then show that presence of path P is necessary for existence of $URMT_{forward}$ protocol in $\mathcal{N}^*_{URMT_{BEF}}$ (Lemma 5, 6 and 7).

Lemma 4 *If there exists a $URMT$ protocol from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}$ tolerating adversary $\mathcal{A}$ then there exists a $URMT_{forward}$ protocol from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}^*_{URMT_{BEF}}$ tolerating adversary $\mathcal{A}^*$.*

Proof: Obvious, since any URMT protocol over $\mathcal{N}$ can be directly run over $\mathcal{N}^*_{URMT_{BEF}}$ without using the newly added temporary nodes at all. □

We next show that presence of path P is necessary for the existence of $URMT_{forward}$ in $\mathcal{N}^*_{URMT_{BEF}}$. Thus we show that for the existence of $URMT_{forward}$ in $\mathcal{N}^*_{URMT_{BEF}}$, there should exist a strong path from $\mathbf{S}$ to $\mathbf{R}$ after deletion of nodes in $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2)) = (B_1 \cup B_2 \cup (F_1 \cap F_2))$ from $\mathcal{N}^*_{URMT_{BEF}}$

(recall that $B'_1 = B_1$, $B'_2 = B_2$ and $(F'_1 \cap F'_2) = (F_1 \cap F_2)$). Define the set $Y \subset \mathbb{P}^*$ as the set of vertices that have a strong path to $\mathbf{R}$ in $\mathcal{N}^*_{URMT_{BEF}}$ that does not use any vertex from $(B_1 \cup B_2 \cup (F_1 \cap F_2))$. Furthermore, let $X = \mathbb{P}^* \setminus (B_1 \cup B_2 \cup (F_1 \cap F_2) \cup Y)$.

Assuming that $\mathbf{S}$ and $\mathbf{R}$ are honest (for otherwise almost any protocol is a valid one for URMT), if the second condition of Theorem 4 fails then $\mathbf{S} \in X$ and $\mathbf{R} \in Y$ (see Fig. 6). Moreover, from the definition of Y, there does not exist edge (x, y) in $\mathcal{N}^*_{URMT_{BEF}}$, such that $x \in X$ and $y \in Y$ (otherwise x also belongs to Y). We first prove the necessity assuming that there does not exist an edge (y, x) in $\mathcal{N}^*_{URMT_{BEF}}$, such that $x \in X$ and $y \in Y$ (Lemma 5). Later, we will show that even if the edge (y, x) exists in $\mathcal{N}^*_{URMT_{BEF}}$, impossibility of $URMT_{forward}$ over $\mathcal{N}^*_{URMT_{BEF}}$ holds, if the second condition of Theorem 4 fails (Lemma 6 and 7).

Lemma 5 *If there is no edge (y, x) in $\mathcal{N}^*_{URMT_{BEF}}$, such that $x \in X$ and $y \in Y$, then the second condition of Theorem 4 is necessary for the existence of $URMT_{forward}$ in $\mathcal{N}^*_{URMT_{BEF}}$ tolerating $\mathcal{A}^*$.*

Proof: Let $\mathcal{N}^*_{URMT_{BEF}}$ does not satisfy the second condition of Theorem 4. Thus in $\mathcal{N}^*_{URMT_{BEF}}$ all the paths between $\mathbf{S}$ and $\mathbf{R}$ pass through the nodes in $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2)) = (B_1 \cup B_2 \cup (F_1 \cap F_2))$ (see Fig. 6). Also let there does not exist an edge (y, x) in $\mathcal{N}^*_{URMT_{BEF}}$, such that vertices $x \in X$ and $y \in Y$. Now suppose there exists a $URMT_{forward}$ protocol Π in $\mathcal{N}^*_{URMT_{BEF}}$. Consider two execution sequence E_1 and E_2 of Π. In E_1, $\mathbf{S}$ wants to send m_1 and the set (B'_1, E'_1, F'_1) is corrupted. In E_2, $\mathbf{S}$ wants to send m_2 and the set (B'_2, E'_2, F'_2) is corrupted. Now using a standard *cut and paste argument* [4, 7, 11, 1], it can be shown that there exists an adversary strategy such that $\mathbf{R}$ can not distinguish with probability more than $\frac{1}{2}$, whether set (B'_1, E'_1, F'_1) is corrupted and the message transmitted is m_1 or the set (B'_2, E'_2, F'_2) is corrupted and the message transmitted is m_2. This contradicts that Π is a $URMT_{forward}$ protocol over $\mathcal{N}^*_{URMT_{BEF}}$. □

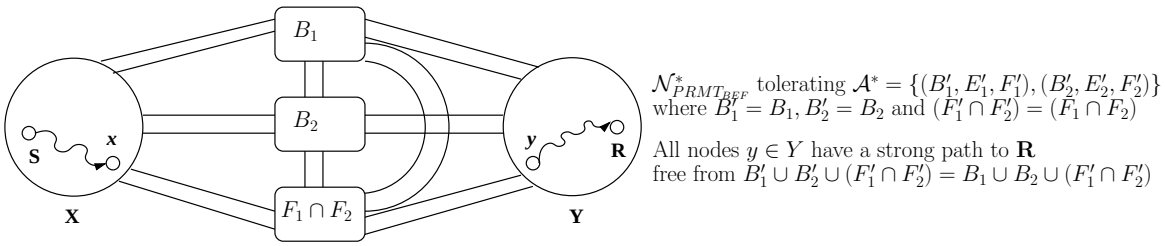


Figure 6: $\mathcal{N}^*_{URMT_{BEF}}$ when every path between $\mathbf{S}$ and $\mathbf{R}$ pass through the nodes in $(B_1 \cup B_2 \cup (F_1 \cap F_2))$

In order to complete the necessity of Theorem 4, we need to prove the impossibility of $URMT_{forward}$ in $\mathcal{N}^*_{URMT_{BEF}}$, even when there exist and edge (y, x) in $\mathcal{N}^*_{URMT_{BEF}}$, such that $y \in Y$ and $x \in X$. Towards that end, we will now prove that every such edge (y, x) in $\mathcal{N}^*_{URMT_{BEF}}$ is essentially "useless", thereby maintaining the impossibility of $URMT_{forward}$ in $\mathcal{N}^*_{URMT_{BEF}}$ as projected by the Lemma 5.

Firstly, how can an edge (y, x) be useful? It can be used by the nodes in Y to send some *secret* messages to the nodes in X, such that the adversary, oblivious of these messages, cannot simulate the messages of X without being distinguished by Y. In other words, x can influence $w \in Y$, by using the data received from y through the edge (y, x). This also implies that there exists a strong path from x to w. So we need to know whether x can really influence w by using the data received from y. If 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 $w \in Y$ (in turn $\mathbf{R}$) and hence can be ignored.

For that, we have to consider whether y is honest or not and also whether $x \in \mathbb{P}$ (the set of physical nodes). By the definition of Y, the node $y \notin (B_1 \cup B_2 \cup (F_1 \cap F_2)$. *Also it is obvious that if $y \in (E_1 \cap E_2)$, then the edge (y, x) cannot be useful, since the adversary can continue to simulate players in X in such a way that players in Y are oblivious of it (since whatever is sent across the edge (y, x) will be fully known to the adversary).* Now in the next lemma, depending on the behavior of y and whether $x \in \mathbb{P}$, we characterize the allowable set(s) of adversary on the strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$. To prove the lemma, we argue that if nodes from some other adversary set(s) (other than those which are specified in Lemma 6) are present on the strong path from x to w, then we can still apply some construction from Table 1 on $\mathcal{N}^*_{URMT_{BEF}}$, contradicting the fact that $\mathcal{N}^*_{URMT_{BEF}}$ is a closure graph of $\mathcal{N}$. Finally, in

Lemma 7, by devising appropriate adversary strategy, we show that the edge (y, x) becomes "useless", if one of the conditions of Lemma 6 is true in $\mathcal{N}^*_{URMT_{BEF}}$, thereby maintaining the necessity of Lemma 5 and hence Theorem 4.

Lemma 6 *Let there be no edges from the nodes in X to the nodes in Y in $\mathcal{N}^*_{URMT_{BEF}}$ after the deletion of nodes from $((B'_1 \cup B'_2 \cup (F'_1 \cap F'_2)))$. If the edge (y, x) exists in $\mathcal{N}^*_{URMT_{BEF}}$, where $y \in Y$ and $x \in X$, then for each $w \in Y$ such that there exists a strong path from y to w (not containing nodes from $(B'_1 \cup B'_2 \cup (F'_1 \cap F'_2)))$, at least one of the following must be true:*

1. *$x \in \mathbb{P}$ (that is, x is a physical node) and at least one of the following holds:*
   (a) *y is honest and there exists an $\alpha \in \{1, 2\}$ such that every strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ passes through some node(s) in $(F'_\alpha \cup B'_\alpha)$ followed by some node(s) in $B'_{\overline{\alpha}}$.*
   (b) *$y \in F'^*_1$ where $F'^*_1 = F'_1 \setminus (F'_1 \cap F'_2)$ and every strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ passes through some node(s) in $B'_2 \cup (F'_1 \cap F'_2)$.*
   (c) *$y \in F'^*_2$ where $F'^*_2 = F'_2 \setminus (F'_1 \cap F'_2)$ and every strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ passes through some node(s) in $B'_1 \cup (F'_1 \cap F'_2)$.*
   (d) *$y \in E'_1$ and every strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ passes through some node(s) in $B'_1 \cup (F'_1 \cap F'_2)$.*
   (e) *$y \in E'_2$ and every strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ passes through some node(s) in $B'_2 \cup (F'_1 \cap F'_2)$.*
2. *$x \notin \mathbb{P}$ (x is a temporary node added by some construction) and letting $z \in \mathbb{P}$ be the first physical node that occurs in every path that emanates from x, at least one of the following holds:*
   (a) *there exists an $\alpha \in \{1, 2\}$ such that every strong path from y to z (starting with the edge (y, x)) in $\mathcal{N}^*_{URMT_{BEF}}$ has a node from F'_α followed by a node from E'_α and there exists $i \in \{1, 2\}$ such that every path from z to w passes through some node(s) in $B'_i \cup (F'_1 \cap F'_2)$.*
   (b) *every strong path from y to z (starting with the edge (y, x)) in $\mathcal{N}^*_{URMT_{BEF}}$ has a node from E'_α and every path from z to w passes through some node(s) in $(B'_\alpha \cup (F'_1 \cap F'_2))$.*

Proof: Since $w \in Y$ and $y \in Y$ has a strong path to w, then according to the property of the nodes in Y, y has a strong path to $\mathbf{R}$ through w, which is free from the nodes in the set $B'_1 \cup B'_2 \cup (F'_1 \cap F'_2)$. Consider the first case where $x \in \mathbb{P}$ and y is honest, then according to condition (1.a), every path from $x \in X$ to $w \in Y$ in $\mathcal{N}^*_{URMT_{BEF}}$ must pass through some node(s) in $(F'_\alpha \cup B'_\alpha)$ followed by some node(s) in $B'_{\overline{\alpha}}$, for some $\alpha \in \{1, 2\}$. If this is not true then y can act as the head of a semi strong path between x and w and using at least one of the constructions from Table 1, it is possible to augment $\mathcal{N}^*_{URMT_{BEF}}$, which is a contradiction. Similarly, using the same argument, it can be shown that if none of the conditions of Lemma 6 are satisfied, then using at least one of the constructions #1, #2, #3 and #4, it is possible to augment $\mathcal{N}^*_{URMT_{BEF}}$ which is a contradiction. This proves the lemma. □

By looking carefully, we find that the conditions in Lemma 6 captures all possible behavior of y, for both the cases when $x \in \mathbb{P}$ and $x \notin \mathbb{P}$. If $x \in \mathbb{P}$, then conditions 1a, 1b, 1c, 1d and 1e enumerates all possibilities of y, since other possibilities of y (say $y \in (F^*_1 \cap E_1)$ etc) will fall under these five enumerations. Similarly if $x \notin \mathbb{P}$, then conditions 2a and 2b captures all possible behavior of the adversary along the path from y to z beginning with the edge (y, x). Each of these conditions (condition 1 and 2) also specifies the corresponding restrictions that the strong paths from x to w (if $x \in \mathbb{P}$) or z to w (if $x \notin \mathbb{P}$) should satisfy. According to Lemma 6, at least one of these conditions (along with the corresponding restriction on the path from x to w or z to w) should be true in $\mathcal{N}^*_{URMT_{BEF}}$. We now show that if this is true, then there exists an adversary strategy, making (y, x) useless.

Lemma 7 *If one of the conditions stated in Lemma 6 is satisfied, then x cannot influence $w \in Y$ (and hence $\mathbf{R}$), using the data received from y along the edge (y, x) in $\mathcal{N}^*_{URMT_{BEF}}$.*

Proof: We consider only the following cases. The remaining cases are similar.

1. *If $x \in \mathbb{P}$ and y is honest, and there exists an $\alpha \in \{1,2\}$ such that every strong path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ passes through some node(s) in $(F'_\alpha \cup B'_\alpha)$ followed by some node(s) in $B'_{\overline{\alpha}}$*: This is the case wherein the Condition 1(a) of Lemma 6 holds (see Fig. 7(a)). In this case, we show that x cannot influence w by considering the following adversarial strategy: If $\overline{\alpha}^{th}$ set in $\mathcal{A}^*$ is corrupted, then the nodes in $B'_{\overline{\alpha}}$ blocks all messages from x to Y (which includes w) and instead send messages obtained by locally simulating the nodes in X (which includes x) assuming that the nodes in F'_α crashed. On the other hand, if the α^{th} set in $\mathcal{A}^*$ is corrupted, then the adversary crashes the nodes from F'_α along the path from x to w, thus ensuring that w is unable to distinguish (with a probability more than half) the two execution scenarios.

2. *$x \in \mathbb{P}$, $y \in F'^*_1$ where $F'^*_1 = F'_1 \setminus (F'_1 \cap F'_2)$ and every path from x to w in $\mathcal{N}^*_{URMT_{BEF}}$ involves node(s) from B'_2*: This is the case wherein the Condition 1(b) of Lemma 6 holds (see Fig. 7(b)). In this case, we show that x cannot influence w by considering the following adversarial strategy: when the second set in $\mathcal{A}^*$ is corrupt, the adversary blocks all the messages from x to Y (which includes w) and instead sends the messages obtained by (locally) simulating the nodes in X (which includes x) assuming that the node y *failed* and sent nothing to x. On the other hand, if the first set in $\mathcal{A}^*$ is corrupt, then the adversary crashes y and thereby ensures that w just cannot distinguish (with a probability more than half) the two execution scenarios.

3. *$x \notin \mathbb{P}$ and every strong path from y to z (starting with the edge (y,x)) in $\mathcal{N}^*_{URMT_{BEF}}$ has a node from E'_α and every path from z to w passes through some node(s) in $(B'_\alpha \cup (F'_1 \cap F'_2))$*: This is the case wherein Condition 2(b) of Lemma 6 holds (see Fig. 7(c)). In this case, if the α^{th} set in $\mathcal{A}^*$ is corrupted, then the adversary will *also know* the data received by z from y (the path from y to z passes through E'_α). Hence, it is easy to see that the adversary can easily simulate the nodes in X which cannot be distinguished by w.

Thus x has no influence on w (and hence $\mathbf{R}$) and can be ignored. Hence the lemma holds. □

From Lemmas 5, 6 and 7, it is proved that conditions of Theorem 4 are necessary for the existence of

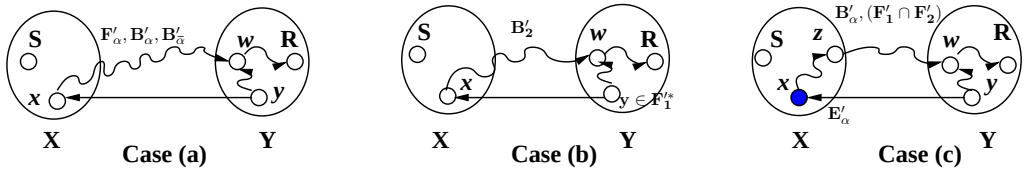


Figure 7: Illustration of the edge (y,x) to be useless in $\mathcal{N}^*_{URMT_{BEF}}$

$URMT_{forward}$ in $\mathcal{N}^*_{URMT_{BEF}}$. The necessity of these conditions for the existence of URMT in $\mathcal{N}$ follows from Lemma 4. This completes the necessity proof of Theorem 4. □

9 APPENDIX V: Illustration of Constructing URMT$_{special}$-BEF-Closure-Digraph $\mathcal{N}^*_{URMT_{BEF}}$

Recall that in subsection 1.3, we have claimed that URMT over $\mathcal{D}$ (shown in Fig. 2 and 8) tolerating $\mathcal{A}_{(1,0,0)}$ (where $t_b = 1$ and $t_p = t_f = 1$) is possible and feasible. Here we show that

(a) $\mathcal{D}$ satisfies the necessary and sufficient conditions for the existence of URMT tolerating $\mathcal{A}_{(1,0,0)}$

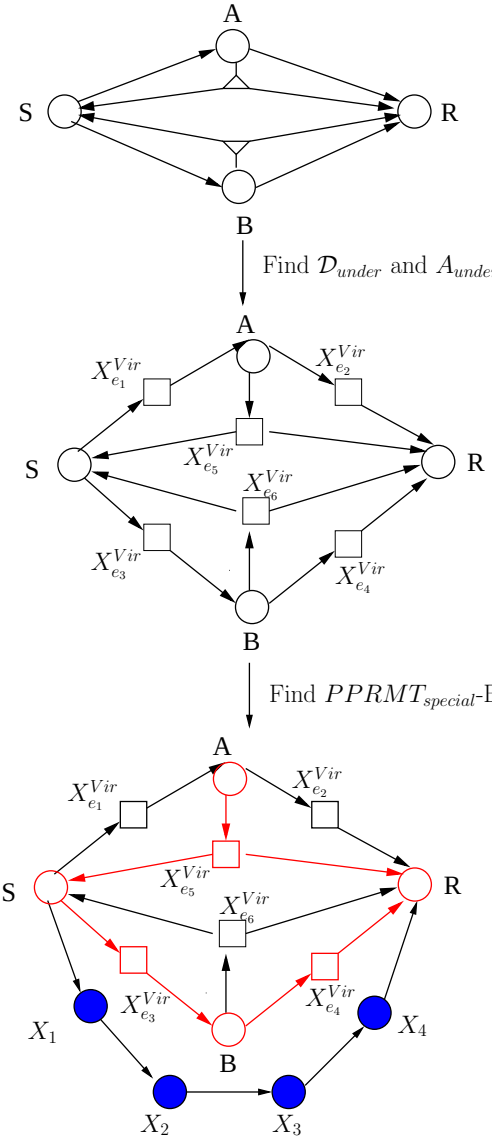
(b) $\mathcal{D}$ does not satisfy the necessary and sufficient condition for tolerating $\mathcal{A}_{(1,1,0)}$.

According to Theorem 1, URMT over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$ is possible iff $URMT_{special}$ over $\mathcal{D}_{under}$ is possible tolerating $\mathbb{A}_{under}$. $\mathcal{D}_{under}$ and $\mathbb{A}_{under}$ corresponding to $\mathcal{D}$ and $\mathcal{A}_{(1,0,0)}$ respectively are shown in Fig. 8. Now according to Theorem 5, $URMT_{special}$ is possible over $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$ iff (1) $\mathcal{D}_{under}$ is such that for each $\alpha \in \{1,2\}$ the deletion of nodes in $((B_\alpha \cup F_\alpha) \setminus \{\mathbf{S}, \mathbf{R}\})$ does not eliminate all the strong paths from $\mathbf{S}$ to $\mathbf{R}$ and (2) The $URMT_{special}$-BEF-Closure-Digraph of $\mathcal{D}_{under}$ with respect to $\mathbb{A}_{under}$ viz., $\mathcal{N}^*_{URMTspl_{BEF}} = (\mathbb{P}^*, \mathbb{E}^*)$ is such that, there exists a strong path from $\mathbf{S}$ to $\mathbf{R}$ in $\mathcal{N}^*_{URMTspl_{BEF}}$, after deleting the vertices from the set $(B'_1 \cup B'_2 \cup (F'_1 \cap$

$F_2')) = (B_1 \cup B_2 \cup (F_1 \cap F_2))$ where the adversary structure closure $\mathcal{A}^*_{spl} = \{(B_1', E_1', F_1'), (B_2', E_2', F_2')\}$. The digraph $\mathcal{D}_{under}$ satisfies condition (1) of Theorem 5 because the strong path from $\mathbf{S}$ to $\mathbf{R}$ after deleting $(B_1 \cup F_1)$ is $(\mathbf{S}, X^{Vir}_{e_3}, B, X^{Vir}_{e_4}, \mathbf{R})$ and the strong path from $\mathbf{S}$ to $\mathbf{R}$ after deleting $(B_2 \cup F_2)$ is $(\mathbf{S}, X^{Vir}_{e_1}, A, X^{Vir}_{e_2}, \mathbf{R})$. Now we construct the $URMT_{special}$-BEF-Closure-Digraph of $\mathcal{D}_{under}$ with respect to $\mathbb{A}_{under}$ viz., $\mathcal{N}^*_{URMTspl_{BEF}}$ (shown in Fig. 8). For that, let $\mathcal{D}_{under} = \mathcal{N}_1$ and $\mathbb{A}_{under} = \mathcal{A}_1$. Now in $\mathcal{N}_1$, consider the nodes $\mathbf{S}$ and $\mathbf{R}$, along with the semi-strong path $(\mathbf{S}, X^{Vir}_{e_5}, \mathbf{R})$ with head $X^{Vir}_{e_5} \in \mathcal{V}$ and strong path $(\mathbf{S}, X^{Vir}_{e_3}, B, X^{Vir}_{e_4}, \mathbf{R})$ from $\mathbf{S}$ to $\mathbf{R}$. The immediate in-neighbor $(\vec{x})$ of head $X^{Vir}_{e_5}$ is node A and the immediate out-neighbor $(\vec{z})$ of head $X^{Vir}_{e_5}$ on the path from the head $X^{Vir}_{e_5}$ to $\mathbf{R}$ is $\mathbf{R}$. The semi-strong path and the strong path, along with $\vec{x}$ and $\vec{z}$, satisfy the conditions of Construction #1 subcase 2 from Table 2. Hence we apply Construction #1 subcase 2 on nodes $\mathbf{S}$ and $\mathbf{R}$ and add four new temporary nodes X_1, X_2, X_3 and X_4 and five new temporary edges $(\mathbf{S}, X_1), (X_1, X_2), (X_2, X_3)$ and (X_3, X_4) in $\mathcal{N}_1$. This augmented digraph is denoted as $\mathcal{N}_2$. The adversary structure $\mathcal{A}_1$ is upgraded such that $F_1 = F_1 \cup \{X_1\},\ F_2 = F_2 \cup \{X_3\}, E_1 = E_1 \cup \{X_2\}, E_2 = E_2 \cup \{X_4\}$. This augmented adversary structure is denoted by $\mathcal{A}_2$. Now notice that no new construction can be applied from Table 1 and 2 on any pair of physical nodes. Hence $URMT_{special}$-BEF-Closure-Digraph of $\mathcal{D}_{under}$ with respect to $\mathbb{A}_{under}$ is $\mathcal{N}^*_{URMTspl_{BEF}} = \mathcal{N}_2$ and the corresponding adversary closure is $\mathcal{A}^*_{spl} = \mathcal{A}_2 = \{(B_1', E_1', F_1'), (B_2', E_2', F_2')\}$. Now in $\mathcal{N}^*_{URMTspl_{BEF}}$, the strong path from $\mathbf{S}$ to $\mathbf{R}$ after deletion of vertices from set $(B_1' \cup B_2' \cup (F_1' \cap F_2')) = (B_1 \cup B_2 \cup (F_1 \cap F_2)) = \{A, B\}$ is $(\mathbf{S}, X_1, X_2, X_3, X_4, \mathbf{R})$. Hence $\mathcal{D}_{under}$ satisfies condition 2 of Theorem 5. So, from Theorem 5 $URMT_{special}$ is possible over $\mathcal{D}_{under}$ tolerating $\mathbb{A}_{under}$ and hence from Theorem 1 URMT is possible over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$.

Now recall the second claim that we made in subsection 1.3: URMT over $\mathcal{D}$ tolerating $\mathcal{A}_{(1,1,0)}$ (where $t_b = t_p = 1$ and $t_f = 1$) is impossible. In this case $\mathbb{A}_{under}$ is represented by it's maximal basis $\overline{\mathbb{A}}_{under} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$ where $(B_1, E_1, F_1) = (\{A\}, \{B, X^{Vir}_{e_1}, X^{Vir}_{e_1}, X^{Vir}_{e_2}, X^{Vir}_{e_3}, X^{Vir}_{e_4}, X^{Vir}_{e_5}, X^{Vir}_{e_6}\}, \emptyset)$ and $(B_2, E_2, F_2) = (\{B\}, \{A, X^{Vir}_{e_1}, X^{Vir}_{e_1}, X^{Vir}_{e_2}, X^{Vir}_{e_3}, X^{Vir}_{e_4}, X^{Vir}_{e_5}, X^{Vir}_{e_6}\}, \emptyset)$. So no construction can be applied between any pair of physical nodes in $\mathcal{D}_{under}$. Hence $URMT_{special}$-BEF-Closure-Digraph of $\mathcal{D}_{under}$ with respect to $\overline{\mathbb{A}}_{under}$ is $\mathcal{D}_{under}$ itself which does not satisfy condition 2 of Theorem 5. Therefore $URMT_{special}$ is impossible over $\mathcal{D}_{under}$ tolerating $\overline{\mathbb{A}}_{under}$ and hence from Theorem 1 URMT is impossible over directed hypergraph $\mathcal{D}$ tolerating $\mathcal{A}_{(1,1,0)}$.

10 APPENDIX D: Modified Constructions when $y \in \mathcal{V}$



Hypergraph $\mathcal{D} = (\mathcal{P}, \mathcal{E})$, where $\mathcal{P} = \{S, A, B, R\}$ and $\mathcal{E} = \{e_1, e_2, e_3, e_4, e_5, e_6\}$ with $e_1 = (S, \{A\})$, $e_2 = (A, \{R\})$, $e_3 = (S, \{B\})$, $e_4 = (B, \{R\})$, $e_5 = (A, \{S, R\})$, $e_6 = (B, \{S, R\})$

$\mathcal{A}_{(1,0,0)}$ where $t_b = 1$ and $t_p = t_f = 0$

$\mathcal{D}_{under} = (\mathcal{P}', \mathcal{E}'), \mathcal{P}' = \mathcal{P} \cup \mathcal{V}$,
$\mathcal{V} = \{X^{Vir}_{e_1}, X^{Vir}_{e_2}, X^{Vir}_{e_3}, X^{Vir}_{e_4}, X^{Vir}_{e_5}, X^{Vir}_{e_6}\}$
$\square$ denotes virtual nodes from the set $\mathcal{V}$

$A_{under} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$ where
$B_1 = \{A\}$ $B_2 = \{B\}$
$E_1 = \{X^{Vir}_{e_1}, X^{Vir}_{e_2}, X^{Vir}_{e_5}\}$ $E_2 = \{X^{Vir}_{e_3}, X^{Vir}_{e_4}, X^{Vir}_{e_6}\}$
$F_1 = \emptyset$ $F_2 = \emptyset$

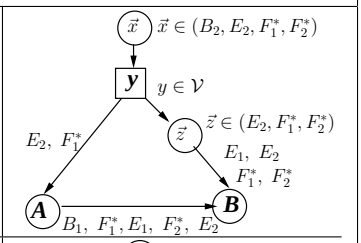
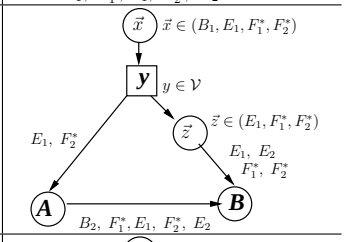
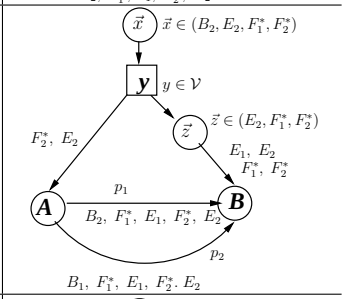
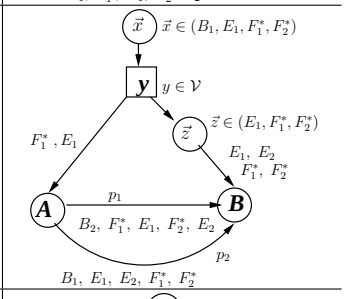
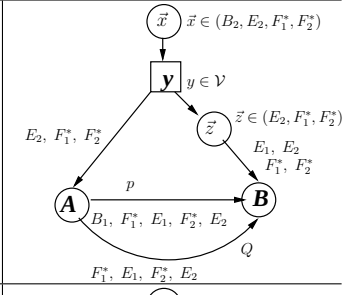
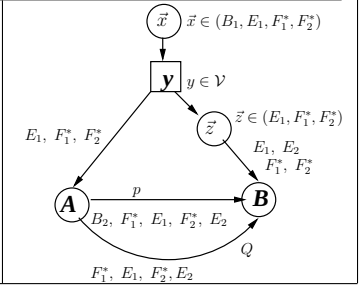
Let $\mathcal{D}_{under} = \mathcal{N}_1$ and $A_{under} = \mathcal{A}_1$

Apply Construction #1 subcase 2 from Table 2 on nodes **S** and **R** where the semi-strong path is $(\mathbf{S}, X^{Vir}_{e_5}, \mathbf{R})$ with head $X^{Vir}_{e_5} \in \mathcal{V}$ and $\vec{x} = A$ and $\vec{z} = \mathbf{R}$
the strong path is $(\mathbf{S}, X^{Vir}_{e_3}, B, X^{Vir}_{e_4}, \mathbf{R})$

$\mathcal{N}_2$,
$\mathcal{A}_2 = \{(B_1, E_1, F_1), (B_2, E_2, F_1)\}$
$B_1 = \{A\}$ $B_2 = \{B\}$
$E_1 = \{X^{Vir}_{e_1}, X^{Vir}_{e_2}, X^{Vir}_{e_5}, X_2\}$ $E_2 = \{X^{Vir}_{e_3}, X^{Vir}_{e_4}, X^{Vir}_{e_6}, X_4\}$
$F_1 = \{X_3\}$ $F_2 = \{X_3\}$

$\mathcal{N}_2 = \mathcal{N}_{PPRMTspl_{BEF}}$ and $\mathcal{A}_2 = \mathcal{A}^*_{spl}$

Figure 8: Illustration of constructing $URMT_{special}$-BEF-Closure-Digraph $\mathcal{N}^*_{URMT_{BEF}}$ of a directed hypergraph $\mathcal{D}$ tolerating $\mathcal{A}_{(1,0,0)}$

No.	Temporary Link	Conditions & Figure	
#1	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in F_1$, $X_2 \in E_1$, $X_3 \in F_2$, $X_4 \in E_2$	1. $y \to A$: $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: Path p: $((B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $\vec{x} \notin (B_1 \cup E_1 \cup (F_1 \cap F_2))$ 5. $\vec{z} \notin (B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2))$	
		1. $y \to A$: $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: Path p: $((B_1 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $\vec{x} \notin (B_2 \cup E_2 \cup (F_1 \cap F_2))$ 5. $\vec{z} \notin (B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2))$	
		1. $y \to A$: $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$ **with the last node from F_2^* before the first node from E_2.** 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: for each $i \in \{1, 2\}$, Path p_i: $(B_i \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $\vec{x} \notin (B_1 \cup E_1 \cup (F_1 \cap F_2))$ 5. $\vec{z} \notin (B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2))$	
		1. $y \to A$: $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$ **with the last node from F_1 before the first node from E_1.** 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$: for each $i \in \{1, 2\}$, Path p_i: $(B_i \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $\vec{x} \notin (B_2 \cup E_2 \cup (F_1 \cap F_2))$ 5. $\vec{z} \notin (B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2))$	
		1. $y \to A$: $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_1) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$, Path p: $(B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $A \to B$, Path Q: $(B_1 \cup B_2 \cup (F_1 \cap F_2)$ 5. $\vec{x} \notin (B_1 \cup E_1 \cup (F_1 \cap F_2))$ 6. $\vec{z} \notin (B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2))$	
		1. $y \to A$: $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_2) \setminus \{A, B\})$ 2. $y \to B$: $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ 3. $A \to B$, Path p: $(B_1 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ **with the last node from F_1^* before the first node from E_1 and the last node from F_2^* before the first node from E_2.** 4. $A \to B$, Path Q: $(B_1 \cup B_2 \cup (F_1 \cap F_2)$ 5. $\vec{x} \notin (B_2 \cup E_2 \cup (F_1 \cap F_2))$ 6. $\vec{z} \notin (B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2))$	

No.	Temporary Link	Conditions & Figure	
#2	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in E_1$, $X_2 \in F_1$, $X_3 \in F_2$, $X_4 \in E_2$	Similar to the construction #1 except that the condition "with the last node from F_1 before the first node from E_1" is removed from the strong path(s) from A to B in all the six cases	Similar to #1 except that first restriction on the ordering of vertices in the strong path(s) from A to B is relaxed
#3	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in F_1$, $X_2 \in E_1$, $X_3 \in E_2$, $X_4 \in F_2$	Similar to the construction #1 except that the condition "with the last node from F_2 before the first node from E_2" is removed from the strong path(s) from A to B in all the six cases	Similar to #1 except that second restriction on the ordering of vertices in the strong path(s) from A to B is relaxed
#4	$A \to X_1 \to X_2 \to X_3 \to X_4 \to B$, $X_1 \in F_1$, $X_2 \in F_2$, $X_3 \in E_1$, $X_4 \in E_2$	Similar to the construction #1 except that both the conditions "with the last node from F_1 before the first node from E_1" and "with the last node from F_2 before the first node from E_2" are removed from the strong path(s) from A to B in all the six cases	Similar to #1 except that both restrictions on the ordering of vertices in the strong path(s) from A to B are relaxed

Table 2: The various constructions (#1 to #4) to convert $\mathcal{N}_{i-1}$ to $\mathcal{N}_i$. In the figures, y denotes the head of semi-strong path q between A and B such that $y \in \mathcal{V}$. $\vec{x}$ denotes the in-neighbor of y and $\vec{z}$ denotes the first physical node on the strong path from y to B. In the figures, the labels along the edges, represents the permissible category of adversary sets in permitted order for the construction. $F_i^* = F_i \setminus (F_1 \cap F_2)$ for $i = \{1, 2\}$. Further in addition to the conditions specified in each construction, the strong path(s) from A to B must satisfy: for each $i \in \{1, 2\}$, every occurrence of a node from $(B_i \cup F_i) \setminus \{A, B\}$ is after the last occurrence of a node from $B_{\bar{i}} \setminus \{A, B\}$, where if $i = 1$ ($i = 2$), then $\bar{i} = 2$ ($\bar{i} = 1$). The constructions are similar to the constructions of Table 1, except that extra restrictions are imposed on $\vec{x}$ and $\vec{z}$.