# Unconditionally Secure Message Transmission in Arbitrary Directed Synchronous Networks Tolerating Generalized Mixed Adversary

Kannan Srinathan[*] Arpita Patra[†‡] Ashish Choudhary[§ †]
C. Pandu Rangan[†||]

## ABSTRACT

In this paper, we re-visit the problem of *unconditionally secure message transmission* (USMT) from a sender **S** to a receiver **R**, who are part of a distributed synchronous network, modeled as an *arbitrary* directed graph. Some of the intermediate nodes between **S** and **R** can be under the control of the adversary having *unbounded* computing power. Desmedt and Wang [4] have given the characterization of USMT in directed networks. However, in their model, the underlying network is abstracted as directed node disjoint paths (also called as wires/channels) between **S** and **R**, where the intermediate nodes are oblivious, message passing nodes and perform no other computation. In this work, we first show that the characterization of USMT given by Desmedt et.al [4] does not hold good for *arbitrary* directed networks, where the intermediate nodes perform some computation, beside acting as message forwarding nodes. We then give the *true* characterization of USMT in arbitrary directed networks. As far our knowledge is concerned, this is the first ever *true* characterization of USMT in arbitrary directed networks.

**Categories and Subject Descriptors:** D. 4. 5 [Reliability]: Fault Tolerance

**General Terms:** Theory, Reliability, Security.

**Keywords:** Information Theoretic Security, Error Probability, Mixed Adversary.

## 1. INTRODUCTION

Achieving reliable and secure communication is one of the fundamental problems in distributed computing. In the problem of *unconditionally reliable message transmission* (URMT), a sender **S** and a receiver **R** are part of a distributed synchronous network and are connected through intermediate nodes. **S** wishes to send a message $m$ to **R**, selected from a finite field $\mathbb{F}$, even in the presence of several kinds of faults in the network. The corruption/fault in the network is modeled by a *centralized adversary*, who has *unbounded computing power* and controls the actions of the nodes (except **S** and **R**) under its influence in a colluded fashion. The challenge is to design a protocol, such that at the end of the protocol, **R** outputs $m' = m$ with probability at least $(1 - \delta)$, for arbitrary small $0 < \delta < \frac{1}{2}$. The problem of *unconditionally secure message transmission* (USMT) requires an *additional* constraint that the adversary should not get *any* information about $m$ in *information theoretic* sense. If **S** and **R** are directly connected by a reliable and secure channel, as assumed in generic secure multiparty computation protocols (see [2, 9, 12, 14, 3]), then reliable and secure communication between **S** and **R** is trivial. However, it is impractical to assume the existence of a direct and secure channel between every two nodes in a network. In such a situation, URMT and USMT protocols help to *simulate* an error free and secure channel between **S** and **R**, with very high probability.

### 1.1 Existing Results and Motivation of Our Work

The URMT and USMT problem was first defined and solved by Franklin et.al [8] in undirected synchronous network, tolerating threshold Byzantine adversary. Roughly speaking, if a node is under the control of the adversary in Byzantine fashion, then the adversary completely dictate the actions of the node and make it behave in an arbitrary fashion. In [8] the authors have abstracted the underlying network in the form of multiple bi-directional vertex disjoint paths, also called as *channels*, between **S** and **R**. The problem of URMT and USMT have been studied extensively in the past in the undirected network model (see [8, 11] and their references). Specially in [11], it is shown that allowing a negligible error probability in the reliability of the protocol significantly reduces the communication complexity of the protocol and also the number of interactions between **S** and **R** during the protocol. Hence it is worth to study URMT and USMT problem in other network models and adversarial models.

Modeling the underlying network as a directed graph is

[*] CSTAR, IIIT Hyderabad, Gachibowli India 500032. Email: srinathan@iiit.ac.in

[†] Department of Computer Science and Engineering, IIT Madras, Chennai India 600036. Email: arpita, ashishc@cse.iitm.ernet.in, rangan@iitm.ernet.in.

[‡] Financial Support from Microsoft Research India Acknowledged.

[§] Financial Support from Infosys Technology India Acknowledged.

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

.

well motivated because in practice not every communication channel admits bi-directional communication. For instance, a base-station may communicate to even a far-off hand-held device but the other way round communication may not be possible. In such a scenario, directed graph is appropriate choice for modeling the underlying network. The existing characterization and protocols for URMT and USMT in undirected networks cannot be directly extended for directed networks. The problem of URMT and USMT in directed networks was first studied by Desmedt et.al [4]. Following the approach of [5] and [8], the authors in [4] abstracted the directed network in the form of uni-directional channels, directed either from **S** to **R** or vice-versa. While doing so, the authors assumed that the intermediate nodes between **S** and **R** are just oblivious, message passing nodes and perform no other computation. However, in [13], Srinathan et.al have shown that such an abstraction is incorrect in the context of URMT. That is, if we assume that the intermediate nodes can perform computation beside acting as message forwarding node, then there exist arbitrary directed graphs over which no URMT protocol according to the characterization of [4], even though there exists an URMT protocol in the graph. Thus to characterize URMT in an arbitrary directed graph, we should consider the underlying graph as a whole, instead of abstracting it in the form of directed channels between **S** and **R**. [1] The authors in [13] have also given the *true* characterization of URMT in an arbitrary directed network. But the characterization holds only for URMT and it cannot be trivially extended for USMT. As far our knowledge is concerned, nothing is known in the literature with respect to the *true* characterization of USMT in an arbitrary directed network, which is the main subject of this paper.

To begin with, we give an example of a directed graph where no USMT is possible according to the characterization of [4], even though there exists an USMT protocol! Consider the network shown in Fig. 1. [2] The previously known theorem characterizing the possibility of USMT in directed synchronous networks tolerating threshold Byzantine adversaries is the following:

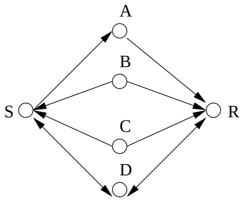


**Figure 1: A directed network**

THEOREM 1 ([4]). *USMT between* **S** *and* **R** *that are connected by uni-directional channels (directed either from* **S** *to* **R** *or from* **R** *to* **S***) tolerating an adversary that can corrupt up to any $t$ channels in Byzantine fashion is possible iff there are at least $2t+1$ disjoint channels between* **S** *and* **R***, of which at least $t+1$ are directed from* **S** *to* **R***.*

If we abstract the network in Fig. 1 in the form of directed channels, then there exists two directed channels between **S** and **R**. Now, if one of the four intermediate nodes among $\{A, B, C, D\}$ is Byzantine corrupted (i.e., $t = 1$), then from Theorem 1, there does not exist any USMT protocol between **S** and **R**. However, if we consider the entire network as a whole and assume that all the intermediate nodes can also perform computation, beside acting as message forwarding nodes, then there indeed exist an USMT protocol tolerating such an adversary.

**USMT protocol for the network in Fig. 1 with $t = 1$:** Let $m \in \mathbb{F}$ be the secret message that **S** wants to send to **R**. Node B selects three random elements $\rho_B^{(1)}, \rho_B^{(2)}, \rho_B^{(3)}$ from $\mathbb{F}$ and sends them to both **S** and **R**. Similarly, node C selects three random elements $\rho_C^{(1)}, \rho_C^{(2)}, \rho_C^{(3)}$ from $\mathbb{F}$ and sends them to both **S** and **R**. Note that if a node is Byzantine corrupted, then it can send different values to **S** and **R**. Let **S** receive the values $\mathbf{S}_B^{(1)}, \mathbf{S}_B^{(2)}, \mathbf{S}_B^{(3)}$ and $\mathbf{S}_C^{(1)}, \mathbf{S}_C^{(2)}, \mathbf{S}_C^{(3)}$ from $B$ and $C$ respectively. Note that if $X \in \{B, C\}$ is uncorrupted, then $\mathbf{S}_X^{(i)} = \rho_X^{(i)}$. Now **S** forms three polynomials $p_1(x), p_2(x)$ and $p_3(x)$, each of degree one (i.e., straight line equation), such that $p_i(2) = \mathbf{S}_B^{(i)}$ and $p_i(3) = \mathbf{S}_C^{(i)}$, for $1 \leq i \leq 3$. Moreover, **S** computes $p_i(1)$ and $p_i(4)$. Let $K_i = p_i(0)$. **S** sends $p_i(1), 1 \leq i \leq 3$ to **R** through node $A$. Similarly, **S** sends $p_i(4), 1 \leq i \leq 3$ to **R** through node $D$. Finally **S** sends the tuple $((m + K_1), K_2(m + K_1) + K_3)$, to **R** through nodes $A$ and $D$.

We now show that **R** can correctly and securely recover $m$ with very high probability. First note that among the four points on each of the three straight lines $p_1(x), p_2(x)$ and $p_3(x)$, at least three points (on each straight line) are common between **S** and **R**. So from the properties of coding theory [10], **R** can locate as well as correct the one point that does not match with that of S (as per rules of error-correction, four points on a straight line are necessary and sufficient to correct one error). Thus, both **S** and **R** can now agree on the values of $p_1(0) = K_1, p_2(0) = K_2$ and $p_3(0) = K_3$. Since the adversary knows at most one point on each of the three straight lines, $K_1, K_2$ and $K_3$ are information theoretically secure. Now, **R** can recover $m$ as follows: let **R** receives the tuple $(x_A, y_A)$ along the path through $A$ and $(x_D, y_D)$ along the path through $D$. Now, **R** verifies if $y_A \overset{?}{=} K_2x_A + K_3$. If yes, then **R** outputs $x_A - K_1$ as the received message. Else, **R** outputs $x_D - K_1$ as the received message.

Since $K_1, K_2, K_3$ are information theoretically secure and $m$ is completely independent of $K_1, K_2$ and $K_3$, it implies that $m$ is also information theoretically secure. Now similar to the proof of information checking protocol of [12], **R** may accept an incorrect message $m' \neq m$ with error probability of at most $\frac{2}{|\mathbb{F}|}$. Formally, if the adversary wants $m' \neq m$ to be accepted by **R**, then he must control the node $A$ and change the tuple $(x_A, y_A)$ passing through it into $(x'_A, y'_A)$, where $x'_A \neq x_A$, such that at **R**'s end $K_2x'_A + K_3 = y'_A$. If $K_2 \neq 0$, then only one $K_3$ will satisfy the equation for the $K_2$ which is held by **R**. On the other hand, if $K_2 = 0$, then irrespective of the value of $K_3$, the equation will always be satisfied. But the probability that $K_2 = 0$ is $\frac{1}{|\mathbb{F}|}$. This is because $K_2 = p_2(0)$, where $p_2(x)$ is formed by interpolating the points $(2, \mathbf{S}_B^2), (3, \mathbf{S}_C^2)$. So, for any $\mathbf{S}_B^2$, there exists an unique $\mathbf{S}_C^2$ such that resultant $p_2(x)$ has 0 as constant coefficient. Thus the probability that **R** may accept an incorrect message $m' \neq m$ is at most $\frac{2}{|\mathbb{F}|}$.

The above example shows that it is incorrect to abstract a

[1] Note that this does not imply the incorrectness of the characterization of URMT given in [4]. The necessary and sufficient condition for URMT given in [4] is correct if the protocol is run between **S** and **R** over abstracted channels.

[2] The current example is taken from [13], where it is used to show the incorrectness of abstracting an arbitrary digraph in the forms of directed channels between **S** and **R**, in the context of URMT.

directed network in the form of directed channels between **S** and **R**, in the context of USMT. This is because the two *heterogeneous* paths involving the nodes $B$ and $C$ respectively are neglected in the channel-based abstraction, since such channels cannot be assigned any orientation. Thus using channel based abstraction, one cannot hope for a complete characterization of USMT over arbitrary directed graph. [3]

**Our Contribution**: We give the first ever *true* characterization for the possibility of USMT in arbitrary directed network tolerating non-threshold mixed adversary, considering the graph as a whole. The reason for considering non-threshold mixed adversary is two fold: First, being a strict generalization of the threshold adversary model, adopting the non-threshold adversary model helps to strengthen the (im)possibility results proved in this paper. Second, the mixed adversary model is a strict generalization of the non-mixed standard (say Byzantine) adversary model and consequently, several practical scenarios are better captured using mixed adversaries than otherwise. For example, in a typical large network, certain nodes may be strongly protected while certain other nodes may be weakly protected. An adversary can only eavesdrop/fail-stop corrupt a strongly protected node, while he may Byzantine corrupt a weakly protected node. Also, it is better to grade the different types of corruption done by the adversary, rather than treating every kind of corruption as Byzantine fault, as this is an overkill. The reader may refer to [1, 7, 6, 11] for scientific proofs for the same. Note that our characterization of USMT is completely different from the characterization of URMT given in [13]. Moreover, we stress that our characterization of USMT is not a trivial extension of the characterization of URMT given in [13].

## 2. MODEL AND DEFINITIONS

The network is modeled as a directed graph $\mathcal{N} = (\mathbb{P}, \mathbb{E})$, where $\mathbb{P}$ is the set of vertices and $\mathbb{E}$ denotes the set of arcs/edges in $\mathcal{N}$. The system is assumed to be synchronous and any protocol is executed in a sequence of rounds wherein in each round, a node can send new messages to his out-neighbors, receive the messages sent in that round by his in-neighbors and perform some computation on the received messages, in that order. We assume that the network topology is known publicly and hence known to all the nodes in the network. Furthermore, we assume that each intermediate node can perform their own computation, beside acting as message forwarding node.

During a protocol execution, a *centralized* adversary may control a subset of the nodes, excluding **S** and **R**. We distinguish between three types of possible control, viz., fail-stop, passive and Byzantine. An adversary can force a fail-stop corrupted nodes to crash at will, but has no access to the secrets/random coins stored in the fail-stop corrupted node's memory. The adversary has full access to the secrets/random coins stored in the passive corrupted node's memory, but the adversary cannot force them to behave arbitrarily. The adversary may fully control a Byzantine corrupted node and make it (mis)behave in an arbitrary fashion.

The adversary is *non-threshold* and is represented by an adversary structure which is an enumeration of all the possible ŞsnapshotsŤ 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 $(B \cap E \cap F) = \emptyset$, which means that the nodes in the set $B$, $E$ and $F$ can be corrupted in Byzantine, passive and fail-stop fashion respectively. Thus, an adversary structure is a collection of such triplets. 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 $B_2 \subseteq B_1, E_2 \subseteq E_1$ and $F_2 \subseteq F_1$, we have $(B_2, E_2, F_2) \in \mathbb{A}$. Throughout the execution of a protocol, the adversary can corrupt nodes from any *one* element (triplet) of $\mathbb{A}$ in Byzantine, passive and fail-stop fashion respectively. Moreover, **S** and **R** have no information about the triplet before the beginning of the protocol. We assume that the adversary is *adaptive*, who uses all information currently at his disposal, to decide the new nodes for corruption in the subsequent rounds. Furthermore, the adversary is *rushing*, who in a particular round sees all the messages sent to him (by the honest players), before sending his own message(s) of that round. We note that $\mathbb{A}$ can be uniquely represented by listing the elements in its maximal basis $\bar{\mathbb{A}}$ 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 $1 \leq i < k$, $(v_i, v_{i+1}) \in \mathbb{E}$.*

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. For example, the path $(\mathbf{S}, B, \mathbf{R})$ in Fig. 1 is a semi-strong path between* **S** *and* **R**, *with $B$ as the head.*

Notice that any strong path can be viewed as a semi-strong path. For example, though the path $(\mathbf{S}, A, \mathbf{R})$ is a strong path from **S** to **R** in Fig. 1, it can be also viewed as a semi-strong path from **S** to **R**, where **S** is the head of the semi-strong path.

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 **S** and **R**. For a message $m$, let **S** computes $auth(m, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ and sends it to **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

[3] As in the case of URMT, this does not imply the incorrectness of the characterization of USMT given in [4]. The necessary and sufficient condition for USMT given in [4] is correct if the protocol is run between **S** and **R** over abstracted channels.

some other value, then except with an error probability of at most $\frac{1}{|\mathbb{F}|}$, **R** will be able to detect it. More specifically, if **S** has sent $(x_1, y_1) = auth(m, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ along the path and if the adversary has changed $(x_1, y_1)$ into $(x_1', y_1')$, where $(x_1', y_1') \neq (x_1, y_1)$, then **R** will be able to detect the corruption by checking $\mathcal{K}_2 x_1' + \mathcal{K}_3 \overset{?}{=} y_1'$. Since $(x_1', y_1') \neq (x_1, y_1)$, the test will fail, except with an error probability of at most $\frac{1}{|\mathbb{F}|}$. The proof of both the properties is similar to the proof of information checking protocol of [12] and hence is omitted.

## 3. USMT IN DIGRAPHS TOLERATING NON-THRESHOLD ADVERSARY

We now characterize USMT in an arbitrary synchronous directed graph $\mathcal{N}$ tolerating an arbitrary non-threshold adversary $\mathbb{A}$. Working out a direct characterization of USMT tolerating entire $\mathbb{A}$ is highly complex and non-intuitive. Rather it is easy to think of a characterization tolerating small sized subsets from $\mathbb{A}$. In [13], it is shown 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. We now show that same holds in the case of USMT also.

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

Proof (sketch): The only-if direction is obvious. For the if-direction, we now show that if an USMT protocol exists while tolerating every monotone subset $\mathcal{A} \subseteq \mathbb{A}$ such that $|\bar{\mathcal{A}}| = 2$, then one can construct an USMT protocol that tolerates $\mathbb{A}$. Suppose that every monotone subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 2$, is tolerable. Then, 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 triplet $(B_i, E_i, F_i)$) into $\mathcal{A}_1 = \{x_1, x_2\}$, $\mathcal{A}_2 = \{x_2, x_3\}$ and $\mathcal{A}_3 = \{x_1, x_3\}$. Now by our assumption, we have USMT protocols $\Pi_1, \Pi_2$ and $\Pi_3$ to tolerate $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{A}_3$ respectively. We now show how to design USMT protocol $\Pi$ to send a message $m$, tolerating $\overline{\mathcal{A}}$.

It is well known that single phase [4] USMT protocol is possible over $2t_b + 1$ wires (vertex disjoint paths from **S** to **R**), of which $t_b$ could be Byzantine corrupted [11, 4]. For $t_b = 1$, it implies that USMT is achievable over three wires, out of which one could be Byzantine corrupted. Let *USMT_Single* be a single phase USMT protocol which runs over three wires $p_1, p_2$ and $p_3$, of which one could be Byzantine corrupted. Moreover, let *USMT_Single* transmits $\alpha_i$ over $p_i$ for $1 \leq i \leq 3$, to send message $m$. We now run the sub-protocols $\Pi_1, \Pi_2$ and $\Pi_3$ in parallel for transmitting $\alpha_1$, $\alpha_2$ and $\alpha_3$ respectively. Since every element of $\bar{\mathcal{A}}$ belongs to at least two of the three $\mathcal{A}_i$'s, **R** gets the correct information in at least two of the three sub-protocols with very high probability. **R** can now output $m$ performing the same computation, as done in *USMT_Single* tolerating 1-active Byzantine adversary. The correctness and secrecy of this USMT protocol tolerating $\bar{\mathcal{A}}$ follows from the correctness and secrecy of the single phase USMT tolerating 1-active adaptive Byzantine adversary. Therefore we can conclude that USMT is possible tolerating any subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 3$.

Applying the same procedure, we find that if USMT is possible tolerating any subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 3$ then it is also possible to design an USMT protocol tolerating any subset $\mathcal{A}$ of $\mathbb{A}$, such that $|\overline{\mathcal{A}}| = 4$. This is because any $\overline{\mathcal{A}} = \{x_1, x_2, x_3, x_4\}$ (where each $x_i$ is an ordered triplet $(B_i, E_i, F_i)$) can be divided into three subsets, each of size three, such that every element in $\overline{\mathcal{A}}$ occurs in at least two of the subsets. More formally, we can divide $\overline{\mathcal{A}}$ into $\mathcal{A}_1 = \{x_1, x_2, x_3\}$, $\mathcal{A}_2 = \{x_2, x_3, x_4\}$ and $\mathcal{A}_3 = \{x_1, x_3, x_4\}$. Now as in the previous case, we can run three USMT protocols (as shown above, these protocols exists) in parallel, transmitting $\alpha_1$, $\alpha_2$ and $\alpha_3$ tolerating the adversary structures $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{A}_3$ respectively. Since every element of $\mathcal{A}$ belongs to at least two of the three $\mathcal{A}_i$'s, **R** gets the correct information in at least two of the three sub-protocols and hence recovers the message by performing same computation as in single phase USMT tolerating 1-active adaptive Byzantine adversary.

In general, any $\mathcal{A} \subseteq \mathbb{A}$ whose maximal basis $|\bar{\mathcal{A}}|$ is of size $\mu > 3$, can be divided into three subsets each of size $\lceil \frac{2\mu}{3} \rceil$, such that every element of $\bar{\mathcal{A}}$ occurs in at least two of the subsets. The rest now follows from induction. □

Remark 1. *The protocol given as a part of sufficiency proof in Theorem 2 is an inductive protocol and is exponential in the size of $\mathbb{A}$. We leave the issue of designing efficient USMT protocol tolerating $\mathbb{A}$ as an open problem.*

Theorem 2 shows that in order to get a complete characterization of USMT tolerating the entire adversary structure $\mathbb{A}$, it is enough if we characterize USMT 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 USMT TOLERATING $\mathcal{A} \subseteq \mathbb{A}$ WITH $|\bar{\mathcal{A}}| = 2$

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

Theorem 3. *Let $\mathcal{N}$ be a directed network under the influence of non-threshold adversary $\mathbb{A}$. Let $\mathcal{A} \subseteq \mathbb{A}$ such that $|\bar{\mathcal{A}}| = 2$, where $\bar{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$. If for each $\alpha \in \{1, 2\}$, there exists two (not necessarily distinct) strong paths $p_\alpha$ and $q_\alpha$ from **S** to **R** in $\mathcal{N}$, such that the path $p_\alpha$ does not contain nodes from $(B_1 \cup B_2 \cup F_{\overline{\alpha}} \cup E_\alpha) \setminus \{\mathbf{S}, \mathbf{R}\}$ and the path $q_\alpha$ does not contain nodes from $(B_\alpha \cup F_\alpha \cup E_\alpha) \setminus \{\mathbf{S}, \mathbf{R}\}$, then there exists an USMT protocol over $\mathcal{N}$ tolerating $\mathcal{A}$. Here if $\alpha = 1(2)$, then $\overline{\alpha} = 2(1)$.*

Proof: For $\alpha \in \{1, 2\}$, let $F_\alpha^* = F_\alpha \setminus (F_1 \cap F_2)$. Similarly, $E_\alpha^* = E_\alpha \setminus (E_1 \cap E_2)$ and $B_\alpha^* = B_\alpha \setminus (B_1 \cap B_2)$. If the conditions of the theorem are true, then there exist four (not necessarily distinct) strong paths $p_1, p_2, q_1$ and $q_2$ from **S** to **R**, which satisfy the conditions given in Table 1. Consider the following USMT protocol to send a secret message $m$: **S** chooses three random and secret keys $K_1, K_2$ and $K_3$ from $\mathbb{F}$ and computes $(x_1, y_1) = auth(m, K_1, K_2, K_3)$. Furthermore, **S** chooses three more random secret keys $K_1', K_2'$ and $K_3'$ (independent of $K_1, K_2$ and $K_3$) and computes $(x_2, y_2) = auth(m, K_1', K_2', K_3')$. **S** then sends the following to **R**: (a) $K_1, K_2, K_3$ along path $p_1$ (b) $K_1', K_2', K_3'$ along path $p_2$ (c) $(x_1, y_1)$ along path $q_2$ and (d) $(x_2, y_2)$ along path $q_1$. We

[4] where only **S** can communicate to **R**

| Paths | Possible Adversary Sets on the path | Comments |
|---|---|---|
| $p_1$ | $F_1^*$, $E_2^*$ | If the first set in $\bar{\mathcal{A}}$ is corrupt, then $p_1$ can be fail-stop corrupt. If the second set in $\bar{\mathcal{A}}$ is corrupt, then $p_1$ can be passively corrupt. |
| $p_2$ | $F_2^*$, $E_1^*$ | If first set in $\bar{\mathcal{A}}$ is corrupt, then $p_2$ can be passively corrupt. If second set in $\bar{\mathcal{A}}$ is corrupt, then $p_2$ can be fail-stop corrupt. |
| $q_1$ | $B_2^*$, $F_2^*$, $E_2^*$ | If first set in $\bar{\mathcal{A}}$ is corrupt, then $q_1$ is honest. If second set in $\bar{\mathcal{A}}$ is corrupt, then $q_1$ can be Byzantine corrupt. |
| $q_2$ | $B_1^*$, $F_1^*$, $E_1^*$ | If the first set in $\bar{\mathcal{A}}$ is corrupt, then $q_2$ can be Byzantine corrupt. If the second set in $\bar{\mathcal{A}}$ is corrupt, then $q_2$ is honest. |

**Table 1: The Strong Paths from S to R Present According to Theorem 3**

assume that during the transmission of these values, if an honest node (which is not under the control of the adversary) in a path receives either no value or syntactically incorrect value (such as a value outside of $\mathbb{F}$ or in incorrect format) from its predecessor along the path, then the honest node stops the forward communication along the path.

**R** now recovers $m$ as follows:

**<u>Case 1:</u>** Suppose **R** receives complete triplet of keys through $p_1$ and $p_2$. Thus all the keys are correctly received (as $p_1$ and $p_2$ cannot be Byzantine corrupted). Let **R** receives $(x'_1, y'_1)$ along path $q_2$ and $(x'_2, y'_2)$ along path $q_1$. **R** checks $y'_1 \overset{?}{=} K_2x'_1 + K_3$. If the test passes, then **R** accepts $x'_1 - K_1$ as the message and terminate. Else **R** accepts $x'_2 - K'_1$ as the message and terminate. The correctness of the message recovery can be argued as follows: suppose the path $q_2$ is Byzantine corrupt and $(x'_1, y'_1) \neq (x_1, y_1)$. This implies that first set in $\bar{\mathcal{A}}$ is corrupted and hence the adversary knows the keys $K'_1, K'_2, K'_3$ by passively listening $p_2$. But adversary will have no information about $K_1, K_2, K_3$. Since $(x_1, y_1)$ is computed from $K_1, K_2, K_3$, by the property of authentication function, except with probability $\frac{1}{|\mathbb{F}|}$, **R** will detect that $q_2$ is corrupted and correctly recovers $m$ from $(x'_2, y'_2)$ ( $= (x_2, y_2)$). Similar argument holds if $q_1$ is Byzantine corrupt. Note that it is possible that **R** either does not receive any 2-tuple or receives some syntactically incorrect value (such as an error message or some value outside $\mathbb{F}$) along $q_1$ ($q_2$). In this case, **R** can easily identify that second (first) set in $\bar{\mathcal{A}}$ is corrupted. **R** now knows that the 2-tuple received over $q_2$ ($q_1$) is correct and recovers $m$ from it.

**<u>Case 2:</u>** Suppose **R** receives the complete triplet of keys through only the path $p_1$. In this case, **R** knows that second set in $\bar{\mathcal{A}}$ is corrupt and so the pair $(x'_1, y'_1)$ received along $q_2$ is correct. Thus, **R** recovers $m$ by computing $x'_1 - K_1$. Using similar argument, we can show that if **R** receives the complete triplet of keys through only $p_2$, then **R** will correctly recover $m$.

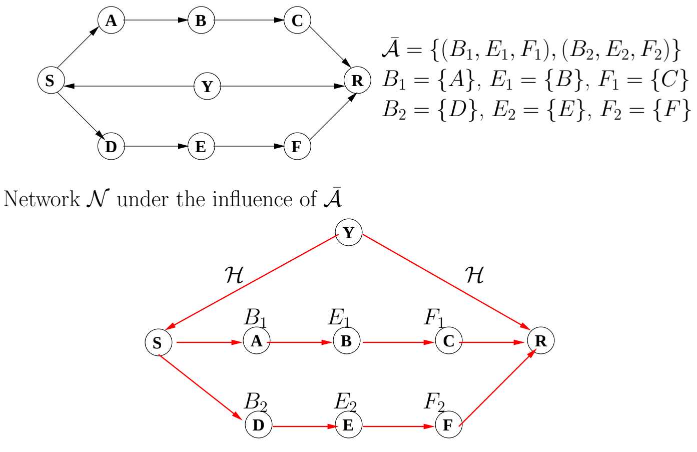
If the first set in $\bar{\mathcal{A}}$ is corrupt, then adversary knows $K'_1, K'_2$ and $K'_3$ and the pair $(x_1, y_1)$ by eavesdropping $p_2$ and $q_2$ respectively. If the second set in $\bar{\mathcal{A}}$ is corrupt, then adversary knows $K_1, K_2$ and $K_3$ and the pair $(x_2, y_2)$ by eavesdropping $p_1$ and $q_1$ respectively. But in both the cases, by the property of authentication function, $m$ is information theoretic secure. □

DEFINITION 5. *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 $p_1, p_2, q_1$ and $q_2$ are present in a network $\mathcal{N}$, then USMT is possible over $\mathcal{N}$. Now the question is whether the physical presence of the paths are necessary in $\mathcal{N}$? The answer is a big no! Here, we show that even in the absence of $p_1, p_2, q_1$ and $q_2$, one can design USMT over $\mathcal{N}$ tolerating $\bar{\mathcal{A}}$, provided the effect of $p_1, p_2, q_1$ and $q_2$ can be simulated over $\mathcal{N}$. This is possible provided $\mathcal{N}$ satisfies certain conditions with respect to $\bar{\mathcal{A}}$.

**<u>Example 1</u>**: Consider the network $\mathcal{N}$ shown in Fig. 2, along with the adversary structure $\bar{\mathcal{A}}$. In $\mathcal{N}$, path $q_1$ = $(\mathbf{S}, D, E, F, \mathbf{R})$ is free from the nodes in $(B_1 \cup E_1 \cup F_1)$, and path $q_2 = (\mathbf{S}, A, B, C, \mathbf{R})$ is free from the nodes in $(B_2 \cup E_2 \cup F_2)$. However, there does not exist any strong path $p_1$ which is free from the nodes in $(B_1 \cup B_2 \cup E_1 \cup F_2)$ and strong path $p_2$ which is free from the nodes in $(B_1 \cup B_2 \cup E_2 \cup F_1)$. So $\mathcal{N}$ does not completely satisfy all the conditions of Theorem 3 with respect to the $\bar{\mathcal{A}}$. However, the effect of $p_1$ and $p_2$ can be simulated in $\mathcal{N}$.


A
B
C
S
Y
R
D
E
F
$\bar{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$
$B_1 = \{A\}$, $E_1 = \{B\}$, $F_1 = \{C\}$
$B_2 = \{D\}$, $E_2 = \{E\}$, $F_2 = \{F\}$
Y
$\mathcal{H}$
$\mathcal{H}$
S
$B_1$
A
$E_1$
B
$F_1$
C
R
$B_2$
D
$E_2$
E
$F_2$
F

Network $\mathcal{N}$ under the influence of $\bar{\mathcal{A}}$

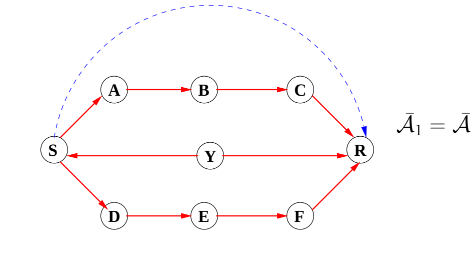
$\mathcal{H}$ - Completely Honest

**Figure 2: A network for Example 1**

Consider the sub-portion of $\mathcal{N}$ with strong paths $(\mathbf{S}, A, B, C, \mathbf{R})$, $(\mathbf{S}, D, E, F, \mathbf{R})$ and semi-strong path $(\mathbf{S}, Y, \mathbf{R})$ (with head $Y$), as shown in the second picture (drawn in red color) in Fig 2. Now consider the following sub-protocol called $\Pi_1^{sim}$ executed over this sub-portion to send a value $s \in \mathbb{F}$ from $\mathbf{S}$ to $\mathbf{R}$: node $Y$ selects three random secret keys $\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3$ from $\mathbb{F}$ and sends to $\mathbf{S}$ and $\mathbf{R}$. Since the semi-strong path is completely honest, both $\mathbf{S}$ and $\mathbf{R}$ correctly receives the same triplet. $\mathbf{S}$ now computes $(x_1, y_1) = auth(s, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ and sends $(x_1, y_1)$ to $\mathbf{R}$, along the strong paths $q_1$ and $q_2$. Let $\mathbf{R}$ receives $(x_1', y_1')$ and $(x_2', y_2')$ along $q_1$ and $q_2$ respectively. $\mathbf{R}$ checks $y_1' \overset{?}{=} \mathcal{K}_2 x_1' + \mathcal{K}_3$. If yes, then $\mathbf{R}$ outputs $x_1' - \mathcal{K}_1$ as $s$ and terminates. Else $\mathbf{R}$ knows that second set in $\bar{\mathcal{A}}$ is corrupted and recovers $s$ by computing $x_2' - \mathcal{K}_1$. It is possible that either $q_1$ or $q_2$ fails to deliver any information to $\mathbf{R}$. However, at least one of them will always correctly deliver $(x_1, y_1)$ to $\mathbf{R}$. If $\mathbf{R}$ does not receive anything along $q_2$, then $\mathbf{R}$ knows that $q_1$ has correctly delivered $(x_1, y_1)$ to $\mathbf{R}$, from which $\mathbf{R}$ can recover $s$. Similar argument holds if $q_2$ does not deliver anything to $\mathbf{R}$. Now irrespective of which set from $\bar{\mathcal{A}}$ is corrupted, the keys will always be oblivious to the adversary as the semi-strong path is honest. So from the property of authentication function, adversary will have *no* information about $s$. Similarly, from the property of authentication function, if $\mathbf{R}$ outputs $s$ from $(x_1', y_1')$, then except with error probability of at most $\frac{1}{|\mathbb{F}|}$, $\mathbf{R}$ outputs correct $s$.

By closely observing protocol $\Pi_1^{sim}$, we find that it has the effect of simulating a direct "virtual edge" between $\mathbf{S}$ and $\mathbf{R}$ with very high probability. So the network $\mathcal{N}$ in Fig. 2 can be enhanced to network $\mathcal{N}_1$ under the influence of $\bar{\mathcal{A}}_1$ as shown in Fig. 3 where in $\mathcal{N}_1$, there exists a "virtual edge" between $\mathbf{S}$ and $\mathbf{R}$ and $\bar{\mathcal{A}}_1 = \bar{\mathcal{A}}$.

Now note that $\mathcal{N}_1$ satisfies the conditions of Theorem 3 with respect to $\bar{\mathcal{A}}_1$, where the virtual edge $(\mathbf{S}, \mathbf{R})$ serves as path $p_1$ and $p_2$. So the USMT protocol $\Pi$ (of Theorem 3) can be executed over $\mathcal{N}_1$ tolerating $\bar{\mathcal{A}}$. However, our goal is to actually design an USMT protocol over $\mathcal{N}$ which is the given physical graph. So we have to simulate the USMT protocol $\Pi$ executed over $\mathcal{N}_1$ tolerating $\bar{\mathcal{A}}$, into an USMT protocol over $\mathcal{N}$ tolerating $\bar{\mathcal{A}}_1$. Our next goal is to demonstrate that simulation.



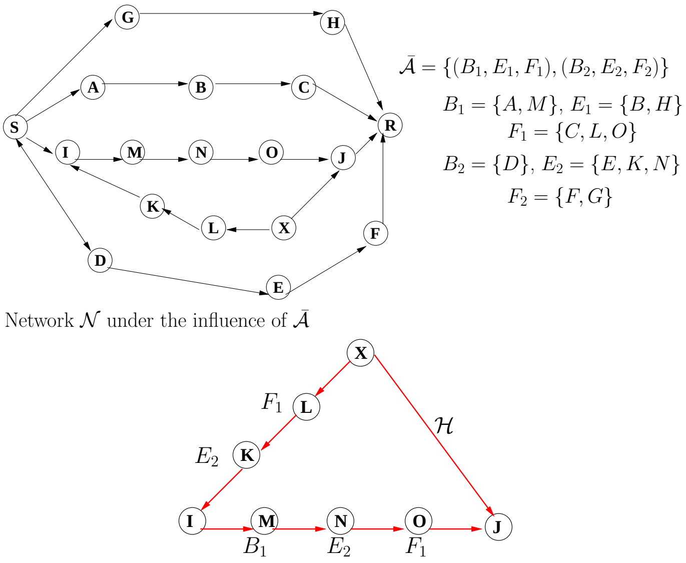
Network $\mathcal{N}_1$ under the influence of $\bar{\mathcal{A}}_1$

**Figure 3: Network $\mathcal{N}$ updated to $\mathcal{N}_1$**

Any value which is sent over $q_1$ or $q_2$ in protocol $\Pi$ 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 edge $(\mathbf{S}, \mathbf{R})$ in protocol $\Pi$ over $\mathcal{N}_1$ can be also sent in $\mathcal{N}$ by using the sub-protocol $\Pi_1^{sim}$. [5] Thus all the steps of $\Pi$ over $\mathcal{N}_1$ can be simulated over $\mathcal{N}$ also. If the error probability of sub-protocol $\Pi_1^{sim}$ is $\delta'$ (which is at most $\frac{1}{|\mathbb{F}|}$), then the error probability of the protocol $\Pi$ simulated over $\mathcal{N}$ is at most $n\delta'$, where $n$ is the number of times sub-protocol $\Pi_1^{sim}$ is executed. So we can make the error probability of resultant USMT protocol over $\mathcal{N}$ to be at most $\delta$, by appropriately selecting $|\mathbb{F}|$ so that $n\delta' = \delta$.

The network in Fig 2 did not satisfy completely the conditions of Theorem 3. Still, there exists a "special structure" in the graph (as shown in the second picture in Fig. 2), due to which the effect of a "simulated virtual edge" between $\mathbf{S}$ and $\mathbf{R}$ could be realized. However, there exists other type of "special structures", which when present in a graph can create the effect of a simulated edge/path. Our next example will demonstrate a "special structure" which can simulate an effect of a "virtual path" rather than an "virtual edge".

<u>**Example 2**</u>: Consider the network $\mathcal{N}$ under the influence of $\bar{\mathcal{A}}$ as shown in Fig. 4. In $\mathcal{N}$, path $q_1 = (\mathbf{S}, D, E, F, \mathbf{R})$ is free from the nodes in $(B_1 \cup E_1 \cup F_1)$, path $q_2 = (\mathbf{S}, A, B, C, \mathbf{R})$ is free from the nodes in $(B_2 \cup E_2 \cup F_2)$ and path $p_2 = (\mathbf{S}, G, H, \mathbf{R})$ is free from the nodes in $(B_1 \cup B_2 \cup E_2 \cup F_1)$. Thus, $\mathcal{N}$ satisfies the conditions of Theorem 3, except that there does not exist any strong path $p_1$ which is free from the nodes in $(B_1 \cup B_2 \cup E_1 \cup F_2)$. However, its effect can be simulated in $\mathcal{N}$.


G
H
A
B
C
S
R
I
M
N
O
J
K
L
X
F
D
E
$\bar{\mathcal{A}} = \{(B_1, E_1, F_1), (B_2, E_2, F_2)\}$
$B_1 = \{A, M\}$, $E_1 = \{B, H\}$
$F_1 = \{C, L, O\}$
$B_2 = \{D\}$, $E_2 = \{E, K, N\}$
$F_2 = \{F, G\}$
X
$F_1$
L
$\mathcal{H}$
$E_2$
K
I
M
N
O
J
$B_1$
$E_2$
$F_1$

Network $\mathcal{N}$ under the influence of $\bar{\mathcal{A}}$

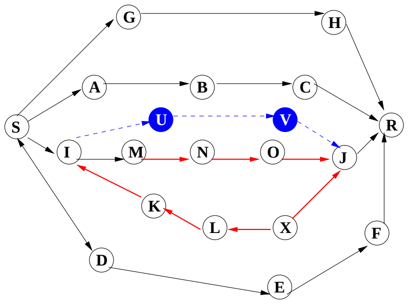
$\mathcal{H}$ - Completely Honest

**Figure 4: A network for Example 2**

Consider the sub-portion of $\mathcal{N}$ involving the strong path $(I, M, N, O, J)$ and semi-strong path $(I, K, L, X, J)$ (with head $X$), as shown in the second picture (drawn in red color) of Fig. 4. Now suppose we execute the following sub-protocol $\Pi_2^{sim}$ over this sub-portion to send a value $s \in \mathbb{F}$ from $I$ to $J$: Node $X$ chooses a non-zero triple $(\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ uniformly at random from $\mathbb{F}^3$ and sends to $I$ and $J$ through the semi-strong path $(I, K, L, X, J)$. Now the following cases may happen: (a) If node $I$ does not receive any triplet from $X$ (this happens if node $L$ gets fail-stop corrupted), then $I$ selects a random triplet $(\mathcal{K}_1', \mathcal{K}_2', \mathcal{K}_3')$ from $\mathbb{F}^3$ on its own, computes $(x_1, y_1) = auth(s, \mathcal{K}_1', \mathcal{K}_2', \mathcal{K}_3')$ and sends to $J$ along the strong path $(I, M, N, O, J)$. (b) If $I$ receives a triple from $X$ (in this case, the triplet is correctly established between $I$ and $J$), then $I$ computes $(x_1, y_1) = auth(s, \mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$ and sends to $J$ along the strong path $(I, M, N, O, J)$.

Since the path from $X$ to $J$ is completely honest, $J$ correctly receives the triplet $(\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3)$. If $J$ either does not

[5] Note that each time an independent random triplet of keys are used to execute the sub-protocol $\Pi_1^{sim}$.



$\bar{\mathcal{A}}_1 = \{(B_1', E_1', F_1'), (B_2', E_2', F_2')\}$

$B_1' = \{A, M\}$, $E_1' = \{B, H\}$

$F_1' = \{C, L, O, U\}$

$B_2' = \{D\}$, $E_2' = \{E, K, N, V\}$

$F_2' = \{F, G\}$

Network $\mathcal{N}_1$ under the influence of $\bar{\mathcal{A}}_1$

**Figure 5: Network $\mathcal{N}$ updated to $\mathcal{N}_1$ and $\bar{\mathcal{A}}$ updated to $\bar{\mathcal{A}}_1$**

receive anything or receives syntactically incorrect values (such as error message or values outside $\mathbb{F}$) along the path $(I, M, N, O, J)$, then it knows that first set in $\bar{\mathcal{A}}$ is corrupted. However, if $J$ receives $(x_1', y_1')$ along the strong path, then $J$ checks $\mathcal{K}_2 x_1' + \mathcal{K}_3 \overset{?}{=} y_1'$. If the test passes, then except with probability $\frac{1}{|\mathbb{F}|}$, $x_1' = x_1$ and $J$ outputs $x_1' - \mathcal{K}_1$ as $s$. On the other hand, if the test fails then $J$ concludes that first set in $\bar{\mathcal{A}}$ is corrupted.

Let us now summarize the effect of sub-protocol $\Pi_2^{sim}$: if the second set in $\bar{\mathcal{A}}$ is corrupted, then the adversary will know the keys $\mathcal{K}_1, \mathcal{K}_2$ and $\mathcal{K}_3$ (by eavesdropping node $K$) and hence $s$ (by eavesdropping node $N$, adversary will know $x_1$). However, $s$ will be correctly received by $J$ (as there are no nodes from $B_2$ along the strong path from $I$ to $J$). On the other hand, if the first set in $\bar{\mathcal{A}}$ is corrupted then the adversary will have no information about $\mathcal{K}_1, \mathcal{K}_2$ and $\mathcal{K}_3$ and hence about $s$. In this case, if $J$ outputs $s$, then with very high probability, it is the correct $s$ and adversary is oblivious of it. However, if $J$ does not output $s$, then it correctly knows that the first set in $\bar{\mathcal{A}}$ is corrupted. Thus sub-protocol $\Pi_2^{sim}$ has the same effect as if there exists a "virtual path" $(I, U, V, J)$, where $U$ and $V$ are "virtual nodes" under the control of $F_1$ and $E_2$ respectively. Note that the ordering of $U$ and $V$ along the virtual path is important. Notice that along the semi-strong path, node $L$ which can be under the control of $F_1$ appears before node $K$, which can be under the control of $E_2$. In sub-protocol $\Pi_2^{sim}$, if the first set in $\bar{\mathcal{A}}$ is corrupted then the adversary can simply block the path from $X$ to $I$. This will simulate the effect as if the virtual path $(I, U, V, J)$ is blocked. On the other hand, if the second set from $\bar{\mathcal{A}}$ is corrupted in $\Pi_2^{sim}$, then the path from $X$ to $I$ will never fail. This will simulate the effect as if the virtual path $(I, U, V, J)$ is not blocked, but is passively controlled by the adversary. If we swap the ordering or $L$ and $K$ along the semi-strong path, then the same sub-protocol $\Pi_2^{sim}$ will have the effect of simulating the virtual path $(I, U, V, J)$, with $U \in E_2$ and $V \in F_1$.

Thus, we can enhance network $\mathcal{N}$ to $\mathcal{N}_1$ and adversary structure $\bar{\mathcal{A}}$ to $\bar{\mathcal{A}}_1$ as shown in Fig. 5, where $\bar{\mathcal{A}}_1$ is same as $\bar{\mathcal{A}}$, except that $F_1' = F_1 \cup \{U\}$ and $E_2' = E_2 \cup \{V\}$. Now $\mathcal{N}_1$ satisfies the conditions of Theorem 3 with respect to $\bar{\mathcal{A}}_1$. Specifically, the paths $q_1, q_2$ and $p_2$ in $\mathcal{N}$ will also be present in $\mathcal{N}_1$. In addition, now the strong path $p_1 = (\mathbf{S}, I, U, V, J, \mathbf{R})$ is free from the nodes in $(B_1' \cup B_2' \cup E_1' \cup F_2')$. So protocol $\Pi$ (of Theorem 3) can be executed over $\mathcal{N}_1$. As demonstrated in **Example 1**, Protocol $\Pi$ which is executed over $\mathcal{N}_1$ can be simulated over $\mathcal{N}$ using $\Pi_2^{sim}$.

Till now, we have demonstrated that the existence of "special structure" in a graph (e.g. the second pictures of Fig 2 and Fig 4 with respect to the adversary structure simulates the effect of virtual edge or path in a given digraph. Let us now pose a very interesting question. Does the simulation of a virtual edge or path between two nodes help to simulate another virtual edge or path between another pair of nodes? Or in other words, is the process of simulation recursive? The answer is yes as shown in next example.

**Example 3**: Consider a network $\mathcal{N}$ which is same as the one shown in the first picture of Fig 4, with the direct edge $(X, J)$ being replaced by the structure between **S** and **R** of Fig 2. That is there are three paths between $X$ and $J$: (i) a completely honest semi-strong path; (ii) a strong path containing three nodes from the sets $B_1$, $E_1$ and $F_1$ respectively in that order; (iii) a strong path containing three nodes from the sets $B_2$, $E_2$ and $F_2$ respectively in that order. The adversary structure $\bar{\mathcal{A}}$ as shown in Fig 4 will be changed accordingly by adding the new nodes in respective sets. From the explanation provided in **Example 2** and by easy observation, we notice that the network $\mathcal{N}$ lacks the existence of path $p_1$ and hence does not satisfy all the conditions of Theorem 3. We now show how to simulate the effect of $p_1$ in $\mathcal{N}$ in the physical absence of $p_1$. Notice that the structure between nodes $X$ and $J$ can simulate a direct virtual edge $(X, J)$ using sub-protocol $\Pi_1^{sim}$ as described in **Example1**. So we may enhance $\mathcal{N}$ to $\mathcal{N}_1$ where the virtual edge $(X, J)$ is included. Similarly, we also enhance adversary structure $\bar{\mathcal{A}}$ to $\bar{\mathcal{A}}_1$, where in this particular case $\bar{\mathcal{A}}_1 = \bar{\mathcal{A}}$. Now network $\mathcal{N}_1$ completely resembles the network shown in the first picture of Fig 4 with the direct physical edge $(X, J)$ replaced by virtual edge $(X, J)$. As explained in **Example 2**, the network $\mathcal{N}_1$ still does not contain path $p_1$. Now the structure between $I$ and $J$ (similar to the one shown in the second picture of Fig 4 with $(X, J)$ being a virtual edge instead of a physical edge) can simulate a virtual path $(I, U, V, J)$ by executing sub-protocol $\Pi_2^{sim}$. The nodes $U$ and $V$ are included in $F_1$ and $E_2$ respectively. With this, we may enhance $\mathcal{N}_1$ to $\mathcal{N}_2$ where the virtual path $(I, U, V, J)$ is included. Now $\mathcal{N}_2$ contains $p_1$ and satisfies all the conditions of Theorem 3. So protocol $\Pi$ of Theorem 3 can be executed over $\mathcal{N}_2$. Now in protocol $\Pi$ any value sent over $p_1$ can be sent by executing $\Pi_2^{sim}$ in $\mathcal{N}_1$. In the protocol executed over $\mathcal{N}_1$, we may need to send some value over the virtual edge $(X, J)$. This can be achieved in original network $\mathcal{N}$ by calling $\Pi_1^{sim}$. Thus the protocol $\Pi$ over $\mathcal{N}_2$ can be simulated over $\mathcal{N}$ with the help of $\Pi_2^{sim}$ and $\Pi_1^{sim}$.

**Summary of the examples**: In **Example 1, 2** and **3**, we have seen networks, which do not satisfy the conditions of Theorem 3, but still protocol $\Pi$ could be simulated on them with very high probability. In **Example 1, 2**, we demonstrated two graphs which contained two different "special structures" (which satisfied some "special properties" with respect to $\bar{\mathcal{A}}$). Those structures lead to the simulation of "virtual edge" and a special type of "virtual path" in the original network. Also, as demonstrated in **Example 3**, the "virtual edge(s)/path(s)" could be added *recursively*. Finally, the enhanced graph, with virtual edge(s)/path(s) added, satisfies the conditions of Theorem 3 and hence we could simulate protocol $\Pi$ on the enhanced graph. But $\Pi$ can be run on the original graph with the help of sub-protocols like $\Pi_1^{sim}$ and $\Pi_2^{sim}$. 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 **USMT-BEF-Closure-Digraph** of the original graph. Finally if **USMT-BEF-Closure-Digraph** satisfies the conditions of Theorem 3, then USMT protocol $\Pi$ exists on the Closure graph. The protocol $\Pi$ can be run on the physical (original) graph using the sub-protocols that simulate the respective virtual edges/paths present in **USMT-BEF-Closure-Digraph**. In the next section, we explore all possibilities of special structures and define **USMT-BEF-Closure-Digraph** formally.

# 5. DEFINITION OF USMT-BEF-CLOSURE-DIGRAPH

DEFINITION 6 (USMT-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* five *different ways by applying one of the constructions from the Tables given in Figure. 6, 8, 9 and 10. Here we provide only Fig. 6. The remaining Tables are provided in* **Appendix D**.

*In the tables, a typical entry as shown in Figure 7 means the following:*

*"In the $n^{th}$ way of construction, we could potentially add a 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 $y \rightarrow A$ which means that the nodes along the strong path from $y$ to $A$ can be completely honest (denoted by* **H***) or may contain nodes from sets $E_2$ and $F_1^*$, where $F_i^* = F_i \setminus (F_1 \cap F_2)$, $i \in \{1, 2\}$.*

REMARK 2. : *A pair of vertices $(A, B)$ may permit at most twenty-four augmentations , corresponding to one of the constructions from Tables given in Figure. 6, 8, 9 and 10. When there is no augmentation 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 USMT-BEF-closure-digraph of $\mathcal{N}$, denoted by $\mathcal{N}^*_{USMT_{BEF}}$ is defined as $\mathcal{N}^*_{USMT_{BEF}} = \mathcal{N}_\nu$. The corresponding adversary structure is defined as $\mathcal{A}^* = \mathcal{A}_\nu$, where $|\bar{\mathcal{A}}^*| = 2$.*

**Example 1, 2** and **3** of previous section demonstrates how to construct USMT-BEF-Closure-Digraph of a given $\mathcal{N}$. In **Example 1**, $\mathcal{N}$ is augmented to $\mathcal{N}_1$ by applying first construction from Construction #1 of Figure 6. In **Example 2**, $\mathcal{N}$ is augmented to $\mathcal{N}_1$ by applying Construction #7 of Figure 8. In **Example 3**, $\mathcal{N}$ is first augmented to $\mathcal{N}_1$ by applying first construction of Construction #1, which is then augmented to $\mathcal{N}_2$ by applying Construction #7. We

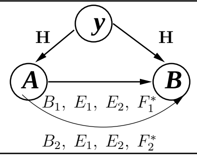
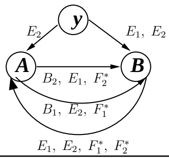
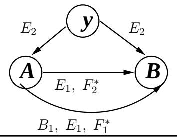
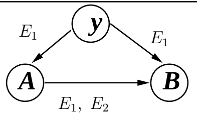
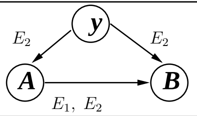
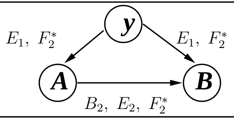
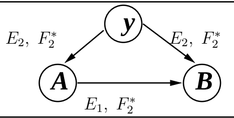
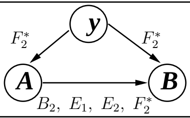
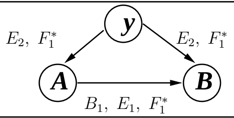
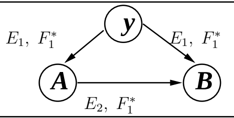
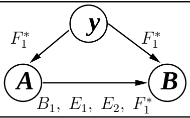
| S.No. | Virtual Link | Conditions & Figures | |
|---|---|---|---|
| #1 | $A \to B$ | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: for each $\alpha \in \{1, 2\}$ avoids nodes from $((B_\alpha \cup F_\alpha) \setminus \{A, B\})$ |   H<br>H<br>$B_1,\ E_1,\ E_2,\ F_1^*$<br>$B_2,\ E_1,\ E_2,\ F_2^*$ |
| | | there exists $\alpha \in \{1, 2\}$ such that<br>• Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_\alpha) \setminus \{A, B\})$<br>• Head $\to B$: $((B_1 \cup B_2 \cup F_1 \cup F_2) \setminus \{A, B\})$<br>• $A \to B$: for each $i \in \{1, 2\}$ avoids nodes from $(B_i \cup F_i \cup E_{\overline{\alpha}}) \setminus \{A, B\})$<br>• $B \to A$: $(B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ | **for** $\alpha = 1$   $E_2$<br>$E_1,\ E_2$<br>$B_2,\ E_1,\ F_2^*$<br>$B_1,\ E_2,\ F_1^*$<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | there exists $\alpha \in \{1, 2\}$ such that<br>• Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_\alpha) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_\alpha) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $(B_1 \cup B_2 \cup F_\alpha \cup E_{\overline{\alpha}}) \setminus \{A, B\})$ and $(B_{\overline{\alpha}} \cup F_{\overline{\alpha}} \cup E_{\overline{\alpha}}) \setminus \{A, B\})$ | **For** $\alpha = 1$  |
| #2 | $A \to X_1 \to B$<br>$X_1 \in E_1$ | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2) \setminus \{A, B\})$ |  |
| #3 | $A \to X_1 \to B$<br>$X_1 \in E_2$ | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup F_2) \setminus \{A, B\})$ |  |
| #4 | $A \to X_1 \to B$<br>$X_1 \in F_2$ | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_1 \cup F_1 \cup E_1) \setminus \{A, B\})$ |  |
| | | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$ |  |
| | | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_1 \cup F_1) \setminus \{A, B\})$ |  |
| #5 | $A \to X_1 \to B$<br>$X_1 \in F_2$ | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$ |  |
| | | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$ |  |
| | | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids nodes from $((B_2 \cup F_2) \setminus \{A, B\})$ |  |

**Figure 6: : Construction for adding virtual path with zero and one intermediate virtual node**

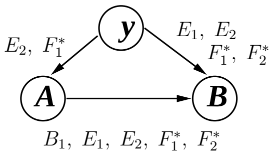
| | | | |
|---|---|---|---|
| #n | $A \to X_1 \to X_2 \to X_3 \to X_4 \to B$ where $X_1 \in F_1$,$X_2 \in F_2$, $X_3 \in E_1$, $X_4 \in E_2$ | • Head $\to A$: avoids nodes from $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$ **with condition** $\mathcal{Q}_1$<br>• Head $\to B$ avoids nodes from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

**Figure 7: : A Typical Entry in Figure. 6, 8, 9 and 10**

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 $\Pi^{sim}$ (as demonstrated in* **Example 1, 2, 3***), which when executed over $\mathcal{N}_{i-1}$ has one of the following outcomes: (a) $\Pi^{sim}$ correctly sends $s$ from $A$ to $B$ over $\mathcal{N}_{i-1}$ with negligible error probability, as demonstrated in* **Example 1***; (b) $\Pi^{sim}$ 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 2***. We do not provide the $\Pi^{sim}$ protocol for every construction given in Figure. 6, 8, 9 and 10 due to space constraint.*

Property 2 (Completeness of the Constructions). *The constructions in Figure. 6, 8, 9 and 10, represents all possible ways of simulating a virtual path between two physical nodes, with zero, one, two, three and four intermediate virtual nodes from $(E_1 \cup F_1 \cup E_2 \cup F_2)$ . We defer the explanation of this property till the proof of Theorem 4.*

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

Proof: The finiteness property follows from the Remark 2 provided in the Definition 6. The proof of the uniqueness property is similar to the proof of Lemma 2 in [13] and hence is omitted[6]. □

Property 3 (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 $\mathcal{N}^*_{USMT_{BEF}}$ has a unique in-neighbor and out-neighbor.*

## 6. TRUE CHARACTERIZATION OF USMT TOLERATING $\mathcal{A}$ WITH $|\bar{\mathcal{A}}| = 2$

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

Theorem 4. *Let $\mathcal{N} = (\mathbb{P}, \mathbb{E})$ be a directed graph, where* **S, 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 $\mathcal{N}^*_{USMT_{BEF}} = (\mathbb{P}^*, \mathbb{E}^*)$ denotes the USMT-BEF-closure-digraph of network $\mathcal{N}$ with respect to $\mathcal{A}$. Moreover, let $\mathcal{N}^*_{USMT_{BEF}}$ 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 USMT between* **S** *and* **R** *is possible in $\mathcal{N}$ tolerating $\mathcal{A}$ iff for each $\alpha \in \{1, 2\}$, the following are true:*

1. *There exists a strong path $\mathcal{P}_\alpha$ from* **S** *to* **R** *in $\mathcal{N}$ avoiding nodes from $(B_\alpha \cup F_\alpha)$.*

2. *There exists two (not necessarily distinct) strong paths $p_\alpha$ and $q_\alpha$ from* **S** *to* **R** *in $\mathcal{N}^*_{USMT_{BEF}}$, such that the path $p_\alpha$ does not contain nodes from $(B'_1 \cup B'_2 \cup F'_{\overline{\alpha}} \cup E'_\alpha)) \setminus \{\mathbf{S}, \mathbf{R}\}$[7] and the path $q_\alpha$ does not contain nodes from $((B'_\alpha \cup F'_\alpha \cup E'_\alpha) \setminus \{\mathbf{S}, \mathbf{R}\})$.*

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. In Theorem 3 the strong paths $p_\alpha$ and $q_\alpha$ are present in the original graph $\mathcal{N}$ and they satisfy certain conditions with respect to $\bar{\mathcal{A}}$, where as in Theorem 4 the same strong paths are present in $\mathcal{N}^*_{USMT_{BEF}}$ and they satisfy the same conditions with respect to $\bar{\mathcal{A}}^*$. Also condition 1 has been added here. Notice that condition 1 has to be satisfied in the original graph $\mathcal{N}$ itself. Now in order to prove the sufficiency of the Theorem 4, we begin with a definition.

Definition 7 ($USMT_{forward}$). *An USMT protocol over digraph $\mathcal{N}_i = (\mathbb{P} \cup \mathbb{V}_i, \mathbb{E} \cup \mathbb{A}_i)$ is called an $USMT_{forward}$ 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 $USMT_{forward}$ protocol over $\mathcal{N}^*_{USMT_{BEF}}$ tolerating $\bar{\mathcal{A}}^*$ (Lemma 2). We then show that if there exists an $USMT_{forward}$ protocol over $\mathcal{N}_i$ for $i > 1$ tolerating $\bar{\mathcal{A}}_i$, then there exists an $USMT_{forward}$ protocol over $\mathcal{N}_{i-1}$ tolerating $\bar{\mathcal{A}}_{i-1}$ (Lemma 3). Now any $USMT_{forward}$ protocol over the original graph $\mathcal{N} = \mathcal{N}_1$ is actually an USMT protocol over $\mathcal{N}$. This is because there are no virtual nodes in $\mathcal{N}$; i.e., $\mathbb{V}_1 = \emptyset$. Since $\mathcal{N}^*_{USMT_{BEF}}$ 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 $USMT_{forward}$ protocol from* **S** *to* **R** *in the network $\mathcal{N}^*_{USMT_{BEF}}$ tolerating the adversary structure $\bar{\mathcal{A}}^*$.*

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

Lemma 3. *For any $i > 1$, there exists an $USMT_{forward}$ protocol from* **S** *to* **R** *in the graph $\mathcal{N}_i$ tolerating the adversary structure $\bar{\mathcal{A}}_i$ if and only if there exists an $USMT_{forward}$ protocol from* **S** *to* **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 $USMT_{forward}$ protocol over $\mathcal{N}_{i-1}$ can be directly run over $\mathcal{N}_i$ *without* using the newly added virtual

[6] In [13], the authors have given the construction of closure graph by considering only Byzantine adversary and fail stop adversary. The constructions given here can be viewed as non-trivial generalization of the constructions given in [13]

[7] If $\alpha = 1(2)$, then $\overline{\alpha} = 2(1)$.

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 $\Pi_i$ be an $USMT_{forward}$ protocol over $\mathcal{N}_i$ tolerating $\bar{\mathcal{A}}_i$. Using $\Pi_i$, we now design an $USMT_{forward}$ protocol $\Pi_{i-1}$ over $\mathcal{N}_{i-1}$ tolerating $\bar{\mathcal{A}}_{i-1}$. Let $\mathcal{I}$ be an instruction in the protocol $\Pi_i$ 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 $\Pi_{i-1}$. 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 finally forwards it to $B$", such that $A, B$ are physical nodes in $\mathcal{N}_i$ (and hence in $\mathcal{N}_{i-1}$) and $X_1$ and $X_2$ are virtual nodes present in $\mathcal{N}_i$ but not in $\mathcal{N}_{i-1}$ (the proof will remain same if $\mathcal{I}$ involves 0,1,2,3 or 4 virtual nodes). In this case, $\mathcal{I}$ cannot be executed over $\mathcal{N}_{i-1}$ directly. But since $X_1$ and $X_2$ 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 $\mathcal{C}$ (from one of the Figure. 6, 8, 9 and 10), to the nodes $A$ and $B$. However, as pointed out in Property 1, for construction $\mathcal{C}$ there is a sub-protocol $\Pi_{\mathcal{C}}^{sim}$ 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. Some examples of sub-protocol $\Pi_{\mathcal{C}}^{sim}$ protocols are provided in **Example 1** and **Example 2** of previous section. Thus, we can replace the instruction $\mathcal{I}$, with sub-protocol $\Pi_{\mathcal{C}}^{sim}$ in $\Pi_{i-1}$. In this way, from $\Pi_i$, we get $\Pi_{i-1}$. 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 $\delta_{sub}$ and the sub-protocol is invoked $N$ times, then the resultant $USMT_{forward}$ protocol is incorrect with a probability up to $N\delta_{sub}$. Since, $\delta_{sub}$ 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 $\delta$ is the tolerance limit of the $USMT_{forward}$ 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 $\mathcal{P}_\alpha$ in $\mathcal{N}$ is obvious. Otherwise all the strong paths from **S** to **R** in $\mathcal{N}$ will contain nodes from $(B_\alpha \cup F_\alpha)$ and the adversary can choose to corrupt the $\alpha^{th}$ set from $\bar{\mathcal{A}}$ and block all the nodes from $(B_\alpha \cup F_\alpha)$, thus refuting any type of communication from **S** to **R** in $\mathcal{N}$. Similarly, the path $q_\alpha$ is necessary in $\mathcal{N}^*_{USMT_{BEF}}$. Otherwise, all the strong paths from **S** to **R** in $\mathcal{N}^*_{USMT_{BEF}}$ will contain nodes from $(B'_\alpha \cup E'_\alpha \cup F'_\alpha)$. In this case, the adversary can choose to corrupt $\alpha^{th}$ set in $\bar{\mathcal{A}}^*$ and block all the nodes from $(B'_\alpha \cup F'_\alpha)$ and eavesdrop all the nodes from $E'_\alpha$. This refutes any type of secure communication from **S** to **R** in $\mathcal{N}^*_{USMT_{BEF}}$ and hence in $\mathcal{N}$.

Finally the necessity of path $p_\alpha$ in $\mathcal{N}^*_{USMT_{BEF}}$ is proved by contradiction. Suppose there exists an USMT protocol $\Pi^*$ in $\mathcal{N}^*_{USMT_{BEF}}$ (and hence in $\mathcal{N}$) tolerating $\bar{\mathcal{A}}^*$ even in the absence of path $p_1$ in $\mathcal{N}^*_{USMT_{BEF}}$ (similar argument is used to show the necessity of $p_2$). Since $p_1$ does not exist, it implies that each of the strong paths from **S** to **R** in $\mathcal{N}^*_{USMT_{BEF}}$ contain nodes from $(B'_1 \cup B'_2 \cup F'_2 \cup E'_1)$. We now divide the set of nodes (virtual + physical) in $\mathcal{N}^*_{USMT_{BEF}}$ as follows: let $Y_1$ be the set of all nodes that have a strong path to **R** in $\mathcal{N}^*_{PPSMT_{BEF}}$ that does not use any vertex from $(B'_1 \cup B'_2 \cup F'_2 \cup E'_1)$. Furthermore, let $X_1 = \mathbb{P}^* \setminus (B'_1 \cup B'_2 \cup F'_2 \cup E'_1) \cup Y_1)$. 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 $p_1$ 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 $p_1$, there always exists an adversary strategy using which $\bar{\mathcal{A}}^*$ can violate the secrecy property of $\Pi^*$ (see Lemma 4 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 USMT (in the absence of $p_1$), thereby maintaining the impossibility of USMT in $\mathcal{N}^*_{USMT_{BEF}}$ as projected by Lemma 4. This is tricky to prove. In order to prove this, we consider the best case for path $p$ i.e. $p$ being honest. We then show that corresponding to this status of $p$, the strong path(s) from $X_1$ to $Y_1$ should definitely satisfy certain properties. If not, then we could augment $\mathcal{N}^*_{USMT_{BEF}}$ by applying at least one of the constructions, thus contradicting the fact that $\mathcal{N}^*_{USMT_{BEF}}$ is **USMT-BEF-Closure-Digraph**. Now once it is shown that the strong path(s) from $X_1$ to $Y_1$ exhibit certain properties, when $p$ is completely honest, we prove that there always exists an adversary strategy which disallows $p$ to help in the possibility of USMT at all.

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

PROPERTY 4 (USEFULNESS OF THE CONSTRUCTIONS). *While the constructions provided in Figure. 6, 8 may directly help in obtaining paths $p_1, p_2, q_1, q_2$ on the closure graph $\mathcal{N}^*_{USMT_{BEF}}$ (as demonstrated in* **Example 1** *and* **Example 2***), the constructions provided in Figure. 9 and 10 may not do so. The reason is that the constructions in Figure. 9 and 10 always add a virtual path containing either nodes of type $F_1$ and $F_2$ or nodes of type $E_1$ and $E_2$. But the properties of $p_1, p_2, q_1$ and $q_2$ says that none of them allow both types of fail-stop or passive nodes. But then there is no reason to consider the constructions in Figure. 9 and 10 as useless. They may be used on intermediate graph as a part of recursion. For example, construction #15 may be applied on an intermediate graph to obtain a new (intermediate) graph with the virtual path $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ where $X_1$, $X_2$ and $X_3$ are included in $F_2$, $E_1$ and $E_2$ respectively. Now this new virtual path may help the current graph to satisfy the conditions of the $3^{rd}$ case of construction #11. Specifically, the new virtual path may act as the strong path from $y$ to $B$ (part of the semi-strong path between $A$ and $B$; see the figure given in Figure 8 corresponding to the $3^{rd}$ entry of construction #11) for $3^{rd}$ case of construction #11 of Figure. 8.*

# 7. CONCLUSION AND OPEN PROBLEMS

We have shown that the existing characterization of USMT in directed networks is inappropriate. We then provided *true* characterization of USMT in arbitrary directed networks. We leave the issue of designing efficient USMT protocols in arbitrary directed networks as an open problem.

# 8. REFERENCES

[1] B. Altmann, M. Fitzi, and U. M. Maurer. Byzantine agreement secure against general adversaries in the dual failure model. In *Proc. of DISC 99*, pages 123–137, 1999.

[2] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *STOC*, pages 1–10, 1988.

[3] David Chaum, Claude Crépeau, and Ivan Damgård. Multiparty unconditionally secure protocols (extended abstract). In *STOC*, pages 11–19, 1988.

[4] Y. Desmedt and Y. Wang. Perfectly secure message transmission revisited. In *Proc. of Advances in Cryptology: Eurocrypt 2002*, LNCS 2332, pages 502–517. Springer-Verlag, 2003.

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

[6] M. Fitzi, M. Hirt, and U. Maurer. Trading correctness for privacy in unconditional multi-party computation. In *Proc. of CRYPTO 98*, pages 121 – 136, 1998.

[7] M. Fitzi, M. Hirt, and U. M. Maurer. General adversaries in unconditional multi-party computation. In *Proc. of ASIACRYPT 99*, pages 232–246, 1999.

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

[9] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *Proc. of 19th ACM STOC*, pages 218–229, 1987.

[10] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error Correcting Codes.* North-Holland Publishing Company, 1978.

[11] A. Patra, A. Choudhary, K. Srinathan, and C. Pandu Rangan. Unconditionally reliable and secure message transmission in undirected synchronous networks tolerating mixed adversary: Possibility, feasibility and optimality. Cryptology ePrint Archive, Report 2008/141, 2008.

[12] Tal Rabin and Michael Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority (extended abstract). In *STOC*, pages 73–85, 1989.

[13] 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.

[14] A. C. Yao. Protocols for secure computations. In *Proc. of 23rd IEEE FOCS*, pages 160–164, 1982.

# APPENDIX C: Necessity Proof of Theorem 4

As mentioned in the necessity proof of Theorem 4, we consider the following two cases and show the impossibility of protocol $\Pi^*$ over $\mathcal{N}^*_{USMT_{BEF}}$ between **S** and **R** in the absence of $p_1$ in both the cases:

1. There does not exist any reverse path(s) from any node(s) in $Y_1$ to any node(s) in $X_1$.
2. There exists reverse path(s) from the nodes in $Y_1$ to the node(s) in $X_1$.

We first consider the case, when there is no reverse path(s) from the node(s) in $Y_1$ to the node(s) in $X_1$ and show the impossibility of USMT protocol $\Pi^*$ between **S** and **R** in the absence of $p_1$.

LEMMA 4. *If there does not exist vertices $x \in X_1$ and $y \in Y_1$ with a strong path from $y$ to $x$ in $\mathcal{N}^*_{USMT_{BEF}}$, then existence of path $p_1$ is necessary for the existence of $\Pi^*$ over the network $\mathcal{N}^*_{USMT_{BEF}}$.*

PROOF: We prove this lemma by contradiction. So assume that USMT protocol $\Pi^*$ exists over $\mathcal{N}^*_{USMT_{BEF}}$, even in the absence of path $p_1$ in $\mathcal{N}^*_{USMT_{BEF}}$. The absence of $p_1$ implies each of the strong paths from **S** to **R** in $\mathcal{N}^*_{USMT_{BEF}}$ contains nodes from $(B'_1 \cup B'_2 \cup F'_2 \cup E'_1)$. So, we may view the network as $X_1$ and $Y_1$ being cut (separated) by four clusters, viz., $B'_1$, $B'_2$, $F'_2$ and $E'_1$. Also, we may assume that these clusters are disjoint (for otherwise the adversary is even more powerful and will not affect this impossibility result). Note that any data flowing from $Y_1$ into any of the clusters has no way of reaching $X_1$ because according to the lemma statement, there is no strong path from the node(s) in $Y_1$ to the node(s) in $X_1$! Thus, the information flowing out of $X_1$ to the clusters is *independent* of what $Y_1$ sent to the clusters. Now, all the information emanating from $X_1$ form four clusters wherein the first cluster is influenced by $B'_1$, the second by $B'_2$ the third by $F'_2$ and the fourth by $E'_1$.

Now in protocol $\Pi^*$ consider the following adversary strategy: (a) if the adversary selects second set from $\bar{\mathcal{A}}^*$ to corrupt, then he blocks all the messages flowing along the nodes of $(B'_2 \cup F'_2)$; (b) if the adversary selects first set from $\bar{\mathcal{A}}^*$ to corrupt, then he passively listen all the messages flowing along the nodes of $(B'_1 \cup E'_1)$. Now since $\Pi^*$ is a valid USMT protocol, **R** can recover message when the adversary corrupts second set from $\bar{\mathcal{A}}^*$. This implies the data passing through the clusters $B'_1$ and $E'_1$ have sufficient information about the message. If so, then adversary can choose first set from $\bar{\mathcal{A}}^*$ and he will also get the secret message by passively listening $B'_1$ and $E'_1$. So, if **R** can recover the message, then so can the adversary. It is easy to note that the communication among the clusters do not help in $\Pi^*$. This is because the adversarial strategy is to block all the messages in the second and the third clusters, contrasted with listening to the first and fourth clusters; clearly communication involving the second and third clusters are useless while the communication between the first and fourth is not going to help since adversary is reading at both ends. This implies that if an USMT protocol between **S** and **R** exists, then there also exists an USMT protocol between another sender **S'** and receiver **R'**, who are connected by four node disjoint paths, where first one in under the control of $B'_1$, the second one is under the control of $B'_2$, the third one is under the control of $F'_2$ and fourth one is under the control of $E'_1$. However, from the results of [11], such an USMT protocol does not exist. Hence $\Pi^*$ is non-existent which is a contradiction. □

We now consider the second case, when there exist strong (reverse) path(s) from $Y_1$ to $X_1$.

LEMMA 5. *Existence of the path $p_1$ is necessary for the existence of USMT protocol $\Pi^*$ over $\mathcal{N}^*_{USMT_{BEF}}$ even if there are strong paths from $Y_1$ to $X_1$.*

PROOF: Suppose there is a node $y \in Y_1$ and another node $x \in X_1$, such that there exists a strong path $p$ from $y$ to $x$. Also consider another node $z$ from $Y_1$ such that $y$ is connected to $z$ through a path not containing nodes from $(B'_1 \cup B'_2 \cup F'_2 \cup E'_1)$. We will prove this lemma considering the best case i.e. considering the case when $p$ is completely honest. Even in this case, we will prove that $\Pi^*$ does not exit, unless path $p_1$ exists. So for all other cases where $p$ contains nodes from some adversary sets, $\Pi^*$ will not exist, in the absence of $p_1$. Hence, let us assume $p$ is completely honest. Also $p_1$ does not exist. So as pointed before, the absence of $p_1$ implies each of the strong paths from $x$ to $z$ in $\mathcal{N}^*_{USMT_{BEF}}$ contains nodes from $(B'_1 \cup B'_2 \cup F'_2 \cup E'_1)$.

Now more specifically, we claim that all the strong path(s) from $x$ to $z$ must pass through the node(s) in $(B'_2 \cup F'_2)$ . We will prove this by contradiction. So assume all the strong path(s) from $x$ to $z$ do not pass through the node(s) in $(B'_2 \cup F'_2)$. Hence they pass through the remaining two sets $(B'_1 \cup E'_1)$. But now we can now apply Construction#7 of Figure. 8 between node $x$ and $z$. Specifically, the semi-strong path and the strong path between $x$ and $z$ satisfy all the conditions of Construction#7: (a) $y$ can act as the head of the semi-strong path between $x$ and $z$, such that the strong path from $y$ to $x$ is completely honest and the strong path from $y$ to $z$ avoids node(s) from $(B'_1 \cup B'_2 \cup F'_2 \cup E'_1)$, (b) the strong path(s) from $x$ to $z$ contains node(s) from $(B'_1 \cup E'_1)$ (i.e. avoids node(s) from $(B'_2 \cup F'_2)$ according to the assumption). So, we can apply Construction #7 from Fig. 8 on $x$ and $z$, which will add the simulated path $x \rightarrow X_1 \rightarrow X_2 \rightarrow z$, where $X_1 \in F'_1$ and $X_2 \in E'_2$. This contradicts the fact that $x \notin Y_1$ and $\mathcal{N}^*_{USMT_{BEF}}$ is an USMT closure graph. Thus, we have shown that in $\mathcal{N}^*_{USMT_{BEF}}$, every path from $x$ to $z$ involves a node from $(B'_2 \cup F'_2)$. Now can the path $p$ be useful for the possibility of USMT? The answer is no! We will prove that $p$ does not help $x$ to influence $z$ in any way. Let the adversary chooses the second set from $\bar{\mathcal{A}}^*$ to corrupt, then his strategy is to block all the information passing through $(B'_2 \cup F'_2)$. So he will block everything from $x$ to $z$. Here, $x$ can essentially help $z$ to identify the corrupted set even without the help of $p$. In other words, in this case if there is an USMT protocol on $\mathcal{N}^*_{USMT_{BEF}}$, then it will exists even in the absence of $p$. Hence $p$ does not help $x$ to influence $z$. This implies the presence of $p$ has no effect on the (im)possibility of $\Pi^*$. So we may assume that $p$ does not exist! This readily proves the necessity of $p_1$ in $\mathcal{N}^*_{USMT_{BEF}}$ from the proof of Lemma 4, when there does not exist any strong path from the set $Y_1$ to $X_1$. □

This completes our proof of the necessity of the path $p_1$ in the Theorem 4. An analogous argument can be given to prove the necessity of the path $p_2$ as well (we do not do it since it is fairly straightforward). This completes the proof of the Theorem 4. □

# APPENDIX D: Tables for Constructions

The various other constructions that are used for constructing **USMT-BEF-Closure** graph are shown in the tables in the following figures. In the figures, we use the following notation with respect to the third column of each entry:

**<u>Notation</u>**: An entry like:

$$\bullet A \rightarrow B \text{ avoids } \mathcal{S}_1 \text{ with}$$

$$\text{last}(\mathcal{S}_2) \text{ precedes first}(\mathcal{S}_3)$$

means the following: there should exist a strong path from $A$ to $B$ avoiding nodes from the set $\mathcal{S}_1$. In addition, the last occurrence of a node from the set $\mathcal{S}_2$ (if they exist) along the strong path should occur before the first occurrence of any node from the set $\mathcal{S}_3$ (if they exist) along the strong path. Thus an entry like

$$\bullet A \rightarrow B \text{ avoids } ((B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A, B\}) \text{ with}$$

$$\text{last}(B_1) \text{ precedes first}(F_2) \text{ and last}(F_1) \text{ precedes first}(E_2)$$

means the following: there should exist a strong path from $A$ to $B$ avoiding nodes from the set $((B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$. In addition, in the strong path, the last occurrence of any node from the set $B_1$ must be before the first occurrence of any node from $F_2$. In addition, along the same strong path, the last occurrence of any node from $F_1$ must be before the first occurrence of any node from $E_2$.

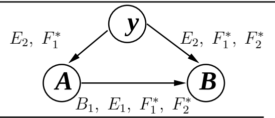
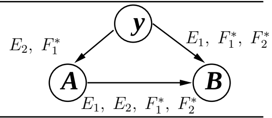
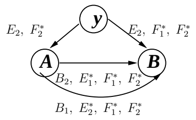
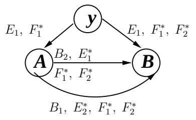
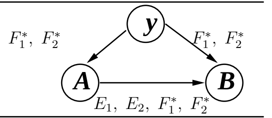
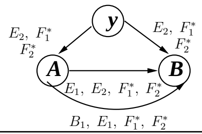
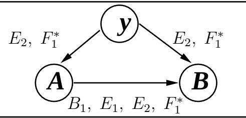
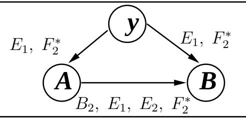
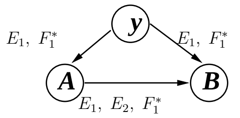
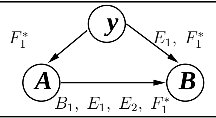
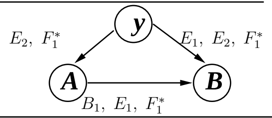
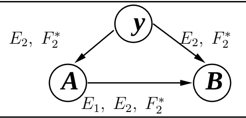
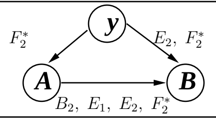
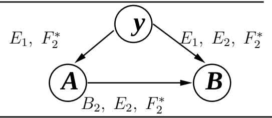
| S.No. | Virtual Link | Conditions & Figures | |
|---|---|---|---|
| #6 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in F_1,\ X_2 \in F_2$ | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2)) \setminus \{A, B\})$<br>• $A \to B$: avoids $((B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ |   $y$<br>$E_2,\ F_1^*$<br>$E_2,\ F_1^*,\ F_2^*$<br>$A$<br>$B$<br>$B_1,\ E_1,\ F_1^*,\ F_2^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$<br>• $A \to B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$ |   $y$<br>$E_2,\ F_1^*$<br>$E_1,\ F_1^*,\ F_2^*$<br>$A$<br>$B$<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$ **with** $E_2$ **after** $F_2$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2)) \setminus \{A, B\})$<br>• $A \to B$: for each $\alpha \in \{1, 2\}$ avoids $((B_\alpha \cup (E_2 \setminus E_\alpha) \cup (F_1 \cap F_2)) \setminus \{A, B\})$ |   $y$<br>$E_2,\ F_2^*$<br>$E_2,\ F_1^*,\ F_2^*$<br>$A$<br>$B$<br>$B_2,\ E_1^*,\ F_1^*,\ F_2^*$<br>$B_1,\ E_2^*,\ F_1^*,\ F_2^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$ **with** $E_1$ **after** $F_1$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A, B\})$<br>• $A \to B$: for each $\alpha \in \{1, 2\}$ avoids $((B_\alpha \cup (E_1 \setminus E_\alpha) \cup (F_1 \cap F_2)) \setminus \{A, B\})$ |   $y$<br>$E_1,\ F_1^*$<br>$E_1,\ F_1^*,\ F_2^*$<br>$A$<br>$B$<br>$B_2,\ E_1^*$<br>$F_1^*,\ F_2^*$<br>$B_1,\ E_2^*,\ F_1^*,\ F_2^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2))$ |   $y$<br>$F_1^*,\ F_2^*$<br>$F_1^*,\ F_2^*$<br>$A$<br>$B$<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | there exists $\alpha \in \{1, 2\}$ such that:<br>• Head $\to A$: avoids $((B_1 \cup B_2 \cup E_\alpha \cup (F_1 \cap F_2)) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup E_\alpha \cup (F_1 \cap F_2)) \setminus \{A, B\})$<br>• $A \to B$: avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2))$ and $((B_{\overline{\alpha}} \cup E_{\overline{\alpha}} \cup (F_1 \cap F_2)) \setminus \{A, B\})$ | **For** $\alpha = 1$<br>  $y$<br>$E_2,\ F_1^*$<br>$F_2^*$<br>$E_2,\ F_1^*$<br>$F_2^*$<br>$A$<br>$B$<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ F_1^*,\ F_2^*$ |
| #7 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in F_1,\ X_2 \in E_2$ | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• $A \to B$: avoids $((B_2 \cup F_2) \setminus \{A, B\})$ |   $y$<br>$E_2,\ F_1^*$<br>$E_2,\ F_1^*$<br>$A$<br>$B$<br>$B_1,\ E_1,\ E_2,\ F_1^*$ |
| #8 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in F_2,\ X_2 \in E_1$ | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids $((B_1 \cup F_1) \setminus \{A, B\})$ |   $y$<br>$E_1,\ F_2^*$<br>$E_1,\ F_2^*$<br>$A$<br>$B$<br>$B_2,\ E_1,\ E_2,\ F_2^*$ |
| #9 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in F_1,\ X_2 \in E_1$ | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids $((B_1 \cup B_2 \cup F_2) \setminus \{A, B\})$ with **last**$(F_1)$ **precedes first**$(E_1)$. |   $y$<br>$E_1,\ F_1^*$<br>$E_1,\ F_1^*$<br>$A$<br>$B$<br>$E_1,\ E_2,\ F_1^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$<br>• $A \to B$: avoids $(B_2 \cup F_2) \setminus \{A, B\})$ with **last**$(B_1 \cup F_1)$ precedes **first**$(E_1)$. |   $y$<br>$F_1^*$<br>$E_1,\ F_1^*$<br>$A$<br>$B$<br>$B_1,\ E_1,\ E_2,\ F_1^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_2) \setminus \{A, B\})$<br>• $A \to B$: avoids $(B_2 \cup F_2 \cup E_2) \setminus \{A, B\})$ with **last**$(B_1 \cup F_1)$ precedes **first**$(E_1)$. |   $y$<br>$E_2,\ F_1^*$<br>$E_1,\ E_2,\ F_1^*$<br>$A$<br>$B$<br>$B_1,\ E_1,\ F_1^*$ |
| #10 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in E_1,\ X_2 \in F_1$ | Same as Construction#9 except that the condition "with **last**$(F_1)$ precedes **first**$(E_1)$" is removed. | Same as Construction#9 |
| #11 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in F_2,\ X_2 \in E_2$ | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$<br>• $A \to B$: avoids from $((B_1 \cup B_2 \cup F_1) \setminus \{A, B\})$ with **last**$(F_2)$ precedes **first**$(E_2)$. |   $y$<br>$E_2,\ F_2^*$<br>$E_2,\ F_2^*$<br>$A$<br>$B$<br>$E_1,\ E_2,\ F_2^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A, B\})$<br>• $A \to B$: avoids $(B_1 \cup F_1) \setminus \{A, B\})$ with **last**$(B_2 \cup F_2)$ precedes **first**$(E_2)$. |   $y$<br>$F_2^*$<br>$E_2,\ F_2^*$<br>$A$<br>$B$<br>$B_2,\ E_1,\ E_2,\ F_2^*$ |
| | | • Head $\to A$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A, B\})$<br>• Head $\to B$: avoids $((B_1 \cup B_2 \cup F_1) \setminus \{A, B\})$<br>• $A \to B$: avoids $(B_1 \cup F_1 \cup E_1) \setminus \{A, B\})$ with **last**$(B_2 \cup F_2)$ precedes **first**$(E_2)$. |   $y$<br>$E_1,\ F_2^*$<br>$E_1,\ E_2,\ F_2^*$<br>$A$<br>$B$<br>$B_2,\ E_2,\ F_2^*$ |
| #12 | $A \to X_1 \to X_2 \to B$<br>$X_1 \in E_2,\ X_2 \in F_2$ | Same as Construction #11 except that the condition "with **last**$(F_2)$ precedes **first**$(E_2)$" is removed. | Same as Construction #11 |

**Figure 8: Constructions for adding virtual path with two intermediate virtual nodes**

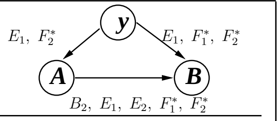
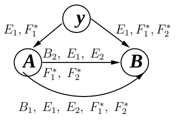
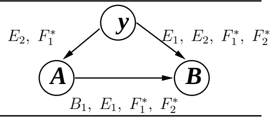
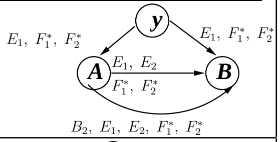
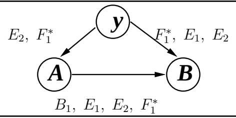
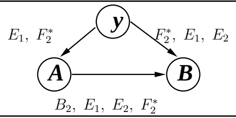
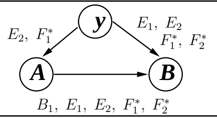
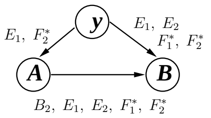
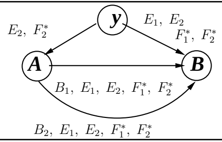
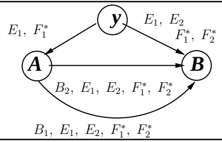
| S.No. | Virtual Link | Conditions & Figures | |
|---|---|---|---|
| #13 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in F_1$, $X_2 \in F_2$, $X_3 \in E_1$ | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_2 \cup F_1) \setminus \{A,B\})$<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $((B_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_2$) precedes first($F_1$)** and **last($F_1$) precedes first($E_1$)**. |  y<br>A<br>B<br>$E_1,\ F_2^*$<br>$E_1,\ F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_2 \cup F_2) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$)**<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: for each $\alpha \in \{1,2\}$, path $p_\alpha$: avoids $((B_\alpha \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_{\bar{\alpha}}$) precedes first($F_\alpha$) and last($F_1$) precedes first($E_1$)**. |  y<br>A<br>B<br>$E_1, F_1^*$<br>$E_1, F_1^*, F_2^*$<br>$B_2,\ E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_1 \cup F_2) \setminus \{A,B\})$<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $(B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_1$) precedes first($F_2$) and last($F_1$) precedes first($E_2$)**. |  y<br>A<br>B<br>$E_2,\ F_1^*$<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ F_1^*,\ F_2^*$ |
| | | • Head → $A, B$: avoids $((B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $(B_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_2$) precedes first($F_1$) and last($F_1$) precedes first($E_1$)**.<br>• $A \rightarrow B$: avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ |  y<br>A<br>B<br>$E_1,\ F_1^*,\ F_2^*$<br>$E_1,\ F_1^*,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| #14 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in F_1$, $X_2 \in E_1$, $X_3 \in E_2$ | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_1 \cup F_2) \setminus \{A,B\})$<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup F_2) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $((B_2 \cup F_2) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$)**. |  y<br>A<br>B<br>$E_2,\ F_1^*$<br>$F_1^*,\ E_1,\ E_2$<br>$B_1,\ E_1,\ E_2,\ F_1^*$ |
| #15 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in F_2$, $X_2 \in E_1$, $X_3 \in E_2$ | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_2 \cup F_1) \setminus \{A,B\})$<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup F_1) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $((B_1 \cup F_1) \setminus \{A,B\})$ with **last($F_2$) precedes first($E_2$)**. |  y<br>A<br>B<br>$E_1,\ F_2^*$<br>$F_2^*,\ E_1,\ E_2$<br>$B_2,\ E_1,\ E_2,\ F_2^*$ |
| #16 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in F_1$, $X_2 \in F_2$, $X_3 \in E_2$ | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_1 \cup F_2) \setminus \{A,B\})$<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $((B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_1$) precedes first($F_2$) and last($F_2$) precedes first($E_2$)**. |  y<br>A<br>B<br>$E_2,\ F_1^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_1 \cup F_1) \setminus \{A,B\})$ with **last($F_2$) precedes first($E_2$)**<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: for each $\alpha \in \{1,2\}$, path $p_\alpha$: avoids $((B_\alpha \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_{\bar{\alpha}}$) precedes first($F_\alpha$) and last($F_2$) precedes first($E_2$)**. |  y<br>A<br>B<br>$E_1,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_2 \cup F_1) \setminus \{A,B\})$<br>• Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $(B_1 \cup E_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_2$) precedes first($F_1$) and last($F_2$) precedes first($E_1$)**. |  y<br>A<br>B<br>$E_2,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A, B$: avoids $((B_1 \cup B_2 \cup E_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$<br>• $A \rightarrow B$: avoids $(B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($B_1$) precedes first($F_2$) and last($F_2$) precedes first($E_2$)**.<br>• $A \rightarrow B$: avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ |  y<br>A<br>B<br>$E_1,\ F_1^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| #17 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in E_1$, $X_2 \in F_1$, $X_3 \in F_2$ | same as construction #13 except that the condition “last($F_1$) precedes first($E_1$) is removed” | same as construction #13 |
| #18 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in F_1$, $X_2 \in E_2$, $X_3 \in F_2$ | same as construction #16 except that the condition “last($F_2$) precedes first($E_2$) is removed” | same as construction #16 |
| #19 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in E_1$, $X_2 \in F_1$, $X_3 \in E_2$ | same as construction #14 except that the condition “last($F_1$) precedes first($E_1$) is removed” | same as construction #14 |
| #20 | $A \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow B$ $X_1 \in E_2$, $X_2 \in F_2$, $X_3 \in E_1$ | same as construction #15 except that the condition “last($F_2$) precedes first($E_2$) is removed” | same as construction #15 |

**Figure 9: Constructions for adding virtual path with three intermediate virtual nodes**

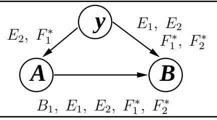
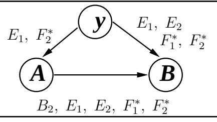
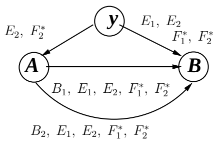
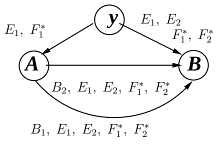
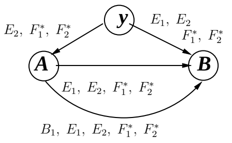
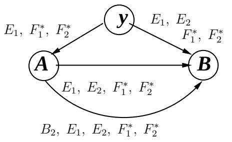
| S.No. | Virtual Link | Conditions & Figures | |
|---|---|---|---|
| #21 | $A \to X_1 \to X_2 \to X_3 \to X_4 \to B$ <br> $X_1 \in F_1,\ X_2 \in F_2,\ X_3 \in E_1,\ X_4 \in E_2$ | • Head → $A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_1) \setminus \{A,B\})$ <br> • Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • $A \to B$: avoids $((B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$) and last($F_2$) precedes first($E_2$).** |   $E_2,\ F_1^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>y<br>A<br>B<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup F_1 \cup E_2) \setminus \{A,B\})$ <br> • Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • $A \to B$: avoids $((B_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($F_2$) precedes first($E_2$) and last($F_1$) precedes first($E_1$).** |   $E_1,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>y<br>A<br>B<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids nodes from $((B_1 \cup B_2 \cup F_1 \cup E_1) \setminus \{A,B\})$ with **last($F_2$) precedes first($E_2$).** <br> • Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • $A \to B$: for each $i \in \{1,2\}$ avoids $(B_i \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$) and last($F_2$) precedes first($E_2$).** |   $E_2,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>y<br>A<br>B<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup F_2 \cup E_2) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$).** <br> • Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • $A \to B$: for each $i \in \{1,2\}$ avoids $(B_i \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **with last($F_1$) precedes first($E_1$) and last($F_2$) precedes first($E_2$).** |   $E_1,\ F_1^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>y<br>A<br>B<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2) \cup E_1) \setminus \{A,B\})$ <br> • Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • $A \to B$, Path $p$: avoids $(B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$) and last($F_2$) precedes first($E_2$)** and avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2))$. <br> • $A \to B$, Path $Q$: avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2)$ |   $E_2,\ F_1^*,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>y<br>A<br>B<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_1,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| | | • Head → $A$: avoids $((B_1 \cup B_2 \cup E_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • Head → $B$: avoids $((B_1 \cup B_2 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ <br> • $A \to B$, Path $p$: avoids $(B_1 \cup (F_1 \cap F_2)) \setminus \{A,B\})$ with **last($F_1$) precedes first($E_1$) and last($F_2$) precedes first($E_2$)** and avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2))$. <br> • $A \to B$, Path $Q$: avoids $(B_1 \cup B_2 \cup (F_1 \cap F_2)$ |   $E_1,\ F_1^*,\ F_2^*$<br>$E_1,\ E_2$<br>$F_1^*,\ F_2^*$<br>y<br>A<br>B<br>$E_1,\ E_2,\ F_1^*,\ F_2^*$<br>$B_2,\ E_1,\ E_2,\ F_1^*,\ F_2^*$ |
| #22 | $A \to X_1 \to X_2 \to X_3 \to X_4 \to B$ <br> $X_1 \in E_1,\ X_2 \in F_1,\ X_3 \in F_2,\ X_4 \in E_2$ | Similar to the construction #21 except that the condition "with last($F_1$) precedes first($E_1$)" is removed from the all the six cases | Similar to #21 except that first restriction on the ordering of vertices in the strong paths from $A$ to $B$ is relaxed |
| #23 | $A \to X_1 \to X_2 \to X_3 \to X_4 \to B$ <br> $X_1 \in F_1,\ X_2 \in E_1,\ X_3 \in E_2,\ X_4 \in F_2$ | Similar to the construction #21 except that the condition "with last($F_2$) precedes first($E_2$)" is removed from the all the six cases | Similar to #21 except that second restriction on the ordering of vertices in the strong paths from $A$ to $B$ is relaxed |
| #24 | $A \to X_1 \to X_2 \to X_3 \to X_4 \to B$ <br> $X_1 \in F_1,\ X_2 \in F_2,\ X_3 \in E_1,\ X_4 \in E_2$ | Similar to the construction #21 except that both the conditions "with last($F_1$) precedes first($E_1$)" and "last($F_2$) precedes first($E_2$)" are removed from the all the six cases | Similar to #21 except that both restrictions on the ordering of vertices in the strong paths from $A$ to $B$ are relaxed |

**Figure 10: Constructions for adding virtual path with four intermediate virtual nodes**