# On Interdependence of Secrets in Collaboration Networks

**Sara Miner More**
*Department of Mathematics and Computer Science*
*McDaniel College, Westminster, Maryland 21157, USA*
*smore@mcdaniel.edu*

**Pavel Naumov**
*Department of Mathematics and Computer Science*
*McDaniel College, Westminster, Maryland 21157, USA*
*pnaumov@mcdaniel.edu*

## Abstract

The paper proposes *Logic of Secrets in Collaboration Networks*, a formal logical system for reasoning about a set of secrets established over a fixed configuration of communication channels. The system's key feature, a multi-channel relation called *independence*, is a generalization of a two-channel relation known in the literature as *nondeducibility*. The main result is the completeness of the proposed system with respect to a semantics of secrets.

## 1 Introduction

Suppose several parties are connected by communication channels that form a network with a fixed topology. In this setting, which we call a *collaboration network*, a pair of parties connected by a channel uses this channel to establish a secret. If the pairs of parties establish their secrets completely independently from other pairs, then possession of one or several of these secrets reveals no information about the other secrets. Assume, however, that secrets are not picked completely independently. Instead, each party with access to multiple channels may enforce some desired interdependency between the secrets it shares with other parties. These "local" interdependencies between secrets known to a single party may result in a "global" interdependency between several secrets, not all of which are known to any single party. Given the fixed topology of the collaboration network, we study what global interdependencies between secrets may exist in the system.

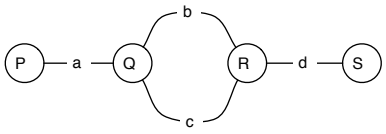


Figure 1: Collaboration network $G_1$.

Consider, for example, the collaboration network depicted in Figure 1. Suppose that the parties collaborate according to the following protocol. Party $P$ picks a random value $a$ from $\{0,1\}$ and sends it to party $Q$. Party $Q$ picks values $b$ and $c$ from $\{0,1\}$ in such a way that $a = b + c \mod 2$ and sends both of these values to $R$. Party $R$ computes $d = b+c \mod 2$ and sends value $d$ to party $S$. In this protocol, it is clear that the values of $a$ and $d$ will always match. We view $a$, $b$, $c$, and $d$ as secrets, conditions $a = b + c \mod 2$ and $d = b + c \mod 2$ as local interdependencies, and condition $a = d \mod 2$ as a global interdependency. Note that in the above example, all channels transmit secret messages in one direction and, thus, the channel network forms a directed graph. However, in the more general setting, two parties might establish the value of a secret through a dialog over their communication channel, with messages traveling in both directions. Thus, in general, we will not assume any specific direction on a channel.

If two or more secrets are not interdependent, then we will say that they are *independent*. (A formal definition of independence will be given in Definition 5.) In the logical system presented in this paper we use independence, not interdependence, as the basic notion simply because it produces a slightly more elegant system. Another way to define independence is to say that secrets are independent if any values of these secrets that can occur in the protocol can also occur simultaneously. For example, secrets $a$ and $b$ in the above protocol are independent, but secrets $a$ and $d$ are not. Furthermore, although secrets $a$, $b$, $c$ in the above protocol are all pairwise independent, the three secrets considered together are not independent.

The independence examples that we have given so far are for a single protocol, subject to a particular set of local interdependencies between secrets. If the topology remains fixed, but the protocol is changed, then secrets which were previously independent could become interdependent, and vice versa. In this paper, however, we study the independence of secrets that follow from the topological structure of the network of channels, no matter which specific protocol is used.

Copyright is held by the author/owner(s).
TARK '09, July 6-8, 2009, California
ISBN: 978-1-60558-560-4...$10.00

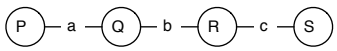


Figure 2: $[a, b] \rightarrow [a, c]$ holds on $G_2$.

For example, it is relatively easy to see that for the graph $G_2$ in Figure 2, if secrets $a$ and $b$ are independent, then secrets $a$ and $c$ are also independent, regardless of the protocol used. This is a property of the network topology, not of the protocol. We say that $[a, b] \rightarrow [a, c]$ is true on topology $G_2$, where $[a, b]$ is our notation for the independence of secrets $a$ and $b$. Another less obvious property of independence is true for graph $G_1$ which defines the network topology in Figure 1, namely, if channels $a$, $b$, and $c$ are independent, then channels $a$ and $d$ are independent: that is, $[a, b, c] \rightarrow [a, d]$ is true on $G_1$. As a final example, consider graph $G_3$ in Figure 3, where the property $[b, c] \rightarrow ([a, e] \rightarrow [a, d])$ holds. In Section 6, we will prove each of these claims.

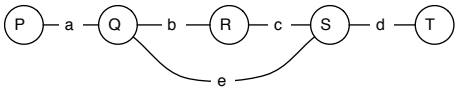


Figure 3: $[a, e] \rightarrow ([b, c] \rightarrow [a, d])$ holds on $G_3$.

In this paper, we present a logic that describes the independence properties of any network topology. The deductive system for this logic operates with binary relation $G \vdash \phi$, where $G$ is a graph that specifies a network topology, and $\phi$ is a propositional statement about secret independence. Our key results are the soundness and completeness of this deductive system with respect to the intended protocol semantics. It is interesting to note that one of the inference rules of this deductive system modifies not only a formula $\phi$, but the graph $G$ as well. The formulas in this logic capture properties of a fixed topology, but the logic itself modifies the topology as part of a derivation. This makes our formal system very different from the traditional deductive systems in mathematical logic.

Our work is related to the study of information flow. Most of the literature in this area, however, studies information flow from the language-based [8, 1] or probabilistic [4, 5] points of view. Historically ([6], page 185), one of the first attempts to capture independence in our sense was undertaken by Goguen and Meseguer [3] through their notion of *noninterference* between two computing devices. Later, Sutherland [9] introduced a *no information flow* relation, which is essentially our independence relation restricted to two-element sets. This relation has since become known in the literature as *nondeducibility*. Cohen [2] presented a related notion called *strong dependence*. Unlike nondeducibility, however, the strong dependence relation is not symmetric. More recently, Halpern and O'Neill [5] introduced $f$-secrecy to reason about multiparty protocols. The $f$-secrecy predicate is a version of nondeducibility that can refer to a value of a certain function of the secret rather than the secret itself. However, all of these works focus on the application of the independence relation in the analysis of secure protocols, whereas the main focus of our work is on logical properties of the relation itself. In a related work [7], we consider a version of the independence relation of the form $[A, B]$, where $A$ and $B$ are not just secrets, but sets of secrets. However, our results in this more general case are restricted to complete graphs.

## 2 Protocol: A Formal Definition

Throughout this paper, we assume a fixed infinite alphabet of variables $a, b, \ldots$, that we refer to as "secret variables". By a network topology we mean a graph whose edges, or "channels", are labeled by secret variables. We allow multiple edges and loops. The set of all channels of graph $G$ will be denoted by $Ch(G)$. One channel may have several labels, but the same label can be assigned to only one channel. Given this, we will often informally refer to "the channel labeled with $a$" as simply "channel $a$".

**Definition 1** *A semi-protocol over a graph $G$ is a pair $\langle V, L\rangle$ such that*

1. *$V(c)$ is an arbitrary set of "values" for each channel $c \in Ch(G)$,*
2. *$L = \{L_p\}_{p \in P}$ is a family of predicates, indexed by parties of the graph $G$, which we call "local conditions". If $c_1, \ldots c_k$ is the list of all channels incident with party $p$, then $L_p$ is a predicate on $V(c_1) \times \cdots \times V(c_k)$.*

**Definition 2** *A run of a semi-protocol $\langle V, L\rangle$ is a function $r$ such that*

1. *$r(c) \in V(c)$ for any channel $c \in Ch(G)$,*
2. *If $c_1, \ldots c_k$ is the list of all channels incident with party $p \in P$, then $L_p(r(c_1), \ldots, r(c_k))$ is true.*

**Definition 3** *A protocol is any semi-protocol that has at least one run.*

The set of all runs of a protocol $\mathcal{P}$ is denoted by $\mathcal{R}(\mathcal{P})$.

**Definition 4** *A protocol $\mathcal{P} = \langle V, L\rangle$ is called finite if the set $V(c)$ is finite for any $c \in Ch(G)$.*

We conclude this section with the key definition of this paper. It is a multi-argument version of Sutherland's binary nondeducibility predicate that we call *independence*.

**Definition 5** *A set of channels $Q = \{q_1, \ldots, q_k\}$ is called independent under protocol $\mathcal{P}$ if for any runs $r_1, \ldots, r_k \in \mathcal{R}(\mathcal{P})$ there is a run $r \in \mathcal{R}(\mathcal{P})$ such that $r(q_i) = r_i(q_i)$ for any $i \in \{1, \ldots, k\}$.*

## 3 Language of Secrets

Informally, by $\Phi(G)$, we denote the set of all properties of secrets in graph $G$. Formally, $\Phi(G)$ is a minimal set defined recursively as follows: (i) for any finite set of secret variables $\{a_1, \ldots, a_n\} \subseteq Ch(G)$, $[a_1, \ldots, a_n] \in \Phi(G)$, (ii) the false constant $\bot \in \Phi(G)$, and (iii) for any formulas $\phi$ and $\psi \in \Phi(G)$, the implication $\phi \rightarrow \psi \in \Phi(G)$. As usual, we assume that conjunction, disjunction, and negation are defined through $\rightarrow$ and $\bot$.

Next, for any graph $G$, we define a relation denoted by $\vDash$. Informally, $\mathcal{P} \vDash \phi$ means that formula $\phi$ is true under protocol $\mathcal{P}$ over graph $G$.

**Definition 6** *For any protocol $\mathcal{P}$ over a graph $G$, and any formula $\phi \in \Phi(G)$ , we define the relation $\mathcal{P} \vDash \phi$ recursively as follows:*

1. *$\mathcal{P} \nvDash \bot$,*
2. *$\mathcal{P} \vDash [a_1, \ldots, a_n]$ if the set of channels $\{a_1, \ldots, a_n\}$ is independent under protocol $\mathcal{P}$,*
3. *$\mathcal{P} \vDash \phi_1 \rightarrow \phi_2$ if $\mathcal{P} \nvDash \phi_1$ or $\mathcal{P} \vDash \phi_2$.*

In this paper, we study the set of formulas that are true under *any* protocol $\mathcal{P}$ as long as graph $G$ remains fixed. The set of all such formulas will be captured by the *Logic of Secrets in Collaboration Networks*. Below, we will list axioms and inference rules for this logic and prove their soundness and completeness.

## 4 Graph Truncation

In preparation for the presentation of an inference rule used in our system, we introduce a graph operation called *truncation*. As usual, a *cut* of a graph is a disjoint partitioning of the nodes of the graph into two sets. A *crossing edge* in a cut is an edge whose ends belong to different sets of the partition. For any set of nodes $X$ of a graph $G$ we use $E(X)$ to denote the set of edges of $G$ whose ends both belong to $X$.

**Definition 7** *Let $G$ be an arbitrary graph and $(X, Y)$ be an arbitrary cut of $G$ (See Figure 4). We define the "truncation" graph $G_X$ of graph $G$ as follows:*

1. *The vertices of graph $G_X$ are the elements of set $X$.*
2. *The edges of $G_X$ are all of the edges from $E(X)$ plus the crossing edges of the cut $(X, Y)$ modified in the following way: if in graph $G$, a crossing edge $c$ connects vertex $v \in X$ with vertex $u \in Y$, then in graph $G_X$, edge $c$ loops from $v$ back into $v$.*

In the remainder of the paper, we refer to vertices as *parties*, edges as *channels*, and crossing edges as *crossing channels*.

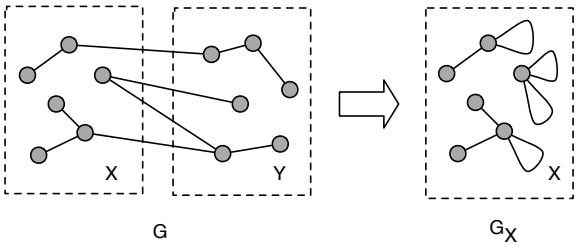


Figure 4: Graph truncation.

## 5 Formal System: Axioms and Rules

We are ready to describe the *Logic of Secrets in Collaboration Networks*. We will write $G \vdash \phi$ to state that formula $\phi \in \Phi(G)$ is provable in this logic. The deductive system for this logic, in addition to propositional tautologies and Modus Ponens inference rule, consists of the *Small Set Axiom*, the *Monotonicity Axiom*, the *Disconnected Partition Axiom*, and the *Truncation Inference Rule*, defined below:

**Small Set Axiom.** Any set that contains less than two elements is independent: $G \vdash [A]$, where $A \subseteq Ch(G)$ and $|A| < 2$.

**Monotonicity Axiom.** Any subset of an independent set of channels is an independent set: $G \vdash [A] \rightarrow [B]$, where $B \subseteq A \subseteq Ch(G)$.

**Disconnected Partition Axiom.** For any partition of the graph $G$ into two disconnected components $X$ and $Y$, if the channels in the $X$-part of $A$ are independent and the channels in the $Y$-part of $A$ are independent, then the whole set $A$ is independent: $G \vdash [A \cap E(X)] \rightarrow ([A \cap E(Y)] \rightarrow [A])$, where $A \subseteq Ch(G)$.

**Truncation Rule.** Let $C \subseteq Ch(G)$ be the set of all crossing channels of partition $(X, Y)$ of graph $G$ and $\phi \in \Phi(G_X)$. If $G_X \vdash \phi$, then $G \vdash [C] \rightarrow \phi$.

## 6 Examples of Proofs

In this section we give examples of proofs in the *Logic of Secrets in Collaboration Networks*. Although proofs in this type of formal system are just sequences of formulas, for the benefit of the reader, we have chosen to connect formulas in these sequences with English sentences.

**Theorem 1** *$G_2 \vdash [a, b] \rightarrow [a, c]$, where $G_2$ is the graph in Figure 2.*

Proof. Graph $G_2'$ (see Figure 5) is a truncation of graph $G_2$ along crossing channels $a$ and $b$. By the Small Set Axiom, $G_2' \vdash [a]$ and $G_2' \vdash [c]$. Hence, by the Disconnected Partition Axiom, $G_2' \vdash [a, c]$. Finally, by the Truncation Rule, $G_2 \vdash [a, b] \rightarrow [a, c]$. ⊠

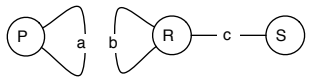


Figure 5: Graph $G_2'$ (shown) is a truncation of graph $G_2$ from Figure 2.

**Theorem 2** *$G_1 \vdash [a, b, c] \rightarrow [a, d]$, where $G_1$ is the graph in Figure 1.*

Proof. Graph $G_1'$ (see Figure 6) is a truncation of graph $G_1$ along crossing channels $a$, $b$, and $c$. By the Small Set Axiom, $G_1' \vdash [a]$ and $G_1' \vdash [d]$. Hence, by the Disconnected Partition Axiom, $G_1' \vdash [a, d]$. Finally, by the Truncation Rule, $G_1 \vdash [a, b, c] \rightarrow [a, d]$. ⊠

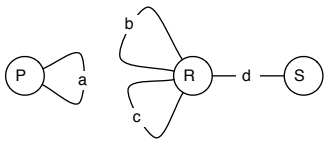


Figure 6: Graph $G_1'$ (shown) is a truncation of graph $G_1$ from Figure 1.

**Theorem 3** *$G_3 \vdash [b, c] \rightarrow ([a, e] \rightarrow [a, d])$, where $G_3$ is the graph in Figure 3.*

Proof. Graph $G_3'$ (see Figure 7) is a truncation of graph $G_3$ along crossing channels $b$ and $c$. Graph $G_3''$ (see Figure 8) is a truncation of graph $G_3'$ along crossing channels $a$ and $e$. By the Small Set Axiom, $G_3'' \vdash [a]$ and $G_3'' \vdash [d]$. Hence, by the Disconnected Partition Axiom, $G_3'' \vdash [a, d]$. Therefore, by the Truncation Rule, $G_3' \vdash [a, e] \rightarrow [a, d]$. Again by the Truncation Rule, $G_3 \vdash [b, c] \rightarrow ([a, e] \rightarrow [a, d])$. ⊠

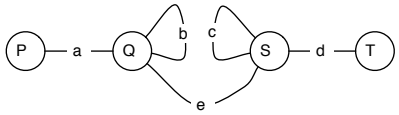


Figure 7: Graph $G_3'$ (shown) is a truncation of graph $G_3$ from Figure 3.

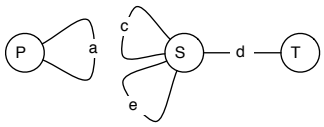


Figure 8: Graph $G_3''$ (shown) is a truncation of graph $G_3'$ from Figure 7.

## 7 Soundness

The proofs of soundness, particularly the soundness of the truncation rule, are non-trivial. For each axiom and inference rule, we provide its justification as a separate theorem.

**Theorem 4 (Small Set)** *For any graph $G$, if $\mathcal{P}$ is an arbitrary protocol over $G$ and any $A \subseteq Ch(G)$ has at most one element, then $\mathcal{P} \vDash [A]$.*

Proof. If $A = \varnothing$, then $\mathcal{P} \vDash [A]$ follows from the existence of at least one run of any protocol. If $A = \{a_1\}$, consider any run $r_1 \in \mathcal{R}(\mathcal{P})$. Pick $r$ to be $r_1$. This guarantees that $r(a_1) = r_1(a_1)$. ⊠

**Theorem 5 (Monotonicity)** *For any graph $G$ and any $A, B \in Ch(G)$ with $B \subseteq A$, if $\mathcal{P} \vDash [A]$, then $\mathcal{P} \vDash [B]$.*

Proof. Let $A = \{a_1, \ldots, a_k\}$, and pick any runs $r_1, \ldots, r_k \in \mathcal{R}(\mathcal{P})$. By the assumption that $\mathcal{P} \vDash [A]$, there is a run $r$ such that $r(a_i) = r_i(a_i)$ for all $i \in \{1, \ldots, k\}$. Since $B \subseteq A$, the claim follows. ⊠

**Theorem 6 (Disconnected Partition)** *For any graph $G$, any partition of $G$ into two disconnected components $X$ and $Y$, and any $A \subseteq Ch(G)$, if $\mathcal{P} \vDash [A \cap E(X)]$ and $\mathcal{P} \vDash [A \cap E(Y)]$, then $\mathcal{P} \vDash [A]$.*

Proof. Let $A \cap E(X) = \{a_1, \ldots, a_k\}$ and $A \cap E(Y) = \{a_{k+1}, \ldots, a_n\}$. Pick any runs $r_1, \ldots, r_n \in \mathcal{R}(\mathcal{P})$. By the assumption of the theorem, there are runs $r^X$ and $r^Y$ such that $r^X(a_i) = r_i(a_i)$ for all $i \in \{1, \ldots, k\}$ and $r^Y(a_i) = r_i(a_i)$ for all $i \in \{k+1, \ldots, n\}$. Define

$$r(c) = \begin{cases} r^X(c) & \text{if } c \in E(X) \\ r^Y(c) & \text{if } c \in E(Y) \end{cases}$$

Note that $r$ satisfies the local conditions at any party $p \in X$ because these conditions are satisfied by $r^X$. Similarly, $r$ satisfies the local conditions at any party $p \in Y$ because these conditions are satisfied by $r^Y$. In addition, $r(a_i) = r_i(a_i)$ for any $i \in \{1, \ldots, n\}$. ⊠

**Theorem 7 (Truncation)** *For any graph $G$, any cut $(X, Y)$ of $G$, and any formula $\phi \in \Phi(G_X)$, if there is a protocol $\mathcal{P}$ over $G$ such that $\mathcal{P} \nvDash \phi$ and $\mathcal{P} \vDash [C]$, where $C$ is the set of all crossing channels of the cut $(X, Y)$, then there is a protocol $\mathcal{P}'$ over the truncation graph $G_X$ such that $\mathcal{P}' \nvDash \phi$.*

Proof. Let $\mathcal{P} = \langle V, L \rangle$. We define a semi-protocol $\mathcal{P}'$ as the pair $\langle V', L' \rangle$, where $V'$ is the restriction of $V$ on $Ch(G_X)$ and local condition $L_p'$ is defined at every party

$p \in X$ as follows: let $b_1, \ldots, b_k$ be the set of all channels incident with party $p$,

$$L'_p(v_1, \ldots, v_k) \equiv \exists r \in \mathcal{R}(\mathcal{P})\ \forall i \in \{1, \ldots, k\}\ (r(b_i) = v_i).$$

First, we will show that $\mathcal{P}'$ has at least one run and, therefore, is a protocol. Indeed, since $\mathcal{P}$ is a protocol, it has a run $r_0$. The restriction of $r_0$ to only the channels in truncated graph $G_X$ clearly satisfies the local conditions $L'_p$. Thus, this restriction is a run of $\mathcal{P}'$.

**Lemma 1** *For any run $r' \in \mathcal{R}(\mathcal{P}')$ there is a run $r \in \mathcal{R}(\mathcal{P})$ such that $r(b) = r'(b)$ for all $b \in Ch(G_X)$.*

Proof. Since $r'$ satisfies the local conditions $L'$, for any $p \in X$ there is a run $r_p \in \mathcal{R}(\mathcal{P})$ such that $r'(b) = r_p(b)$ for any channel $b$ incident with party $p$. For any crossing channel $c \in C$ there is a unique party $p \in X$ such that $c$ is incident with $p$. We will denote this party by $x(c)$. By the assumption of the theorem, $\mathcal{P} \vDash [C]$. Thus, there is a run $\hat{r} \in \mathcal{R}(\mathcal{P})$ such that $\hat{r}(c) = r_{x(c)}(c)$ for any $c \in C$. We are ready to define run $r \in \mathcal{R}(\mathcal{P})$:

$$r(b) = \begin{cases} r'(b) & \text{if } b \in Ch(G_X) \\ \hat{r}(b) & \text{if } b \notin Ch(G_X) \end{cases}$$

By the above definition, $r(b) = r'(b)$ for all $b \in E(X) \cup C$. Thus, we only need to show that $r \in \mathcal{R}(\mathcal{P})$. Indeed, consider any party $p$ in $G$. Let $\{b_1, \ldots, b_n\}$ be the set of all channels incident with $p$. It is sufficient to show that $L_p(r(b_1), \ldots, r(b_n))$.

If $p \notin X$, then $b_1, \ldots, b_n \notin E(X)$. Thus, $r(b_i) = \hat{r}(b_i)$ for any $i \in \{1, \ldots, n\}$. The required condition follows from the fact that $\hat{r} \in \mathcal{R}(\mathcal{P})$.

Suppose that $p \in X$. Note that for any channel $c \in C$ which is incident with $p$, we have $r(c) = \hat{r}(c) = r_{x(c)}(c) = r_p(c)$. On the other hand, if $b$ is incident with $p$ and $b \notin C$, then $r(b) = r'(b) = r_p(b)$. Hence, $r(b) = r_p(b)$ for any $b$ incident with $p$ whether or not $b$ belongs to $C$. The required condition follows from $r_p \in \mathcal{R}(\mathcal{P})$. ⊠

**Lemma 2** *For any set of channels $Q = \{q_1, \ldots, q_n\}$ in graph $G_X$, $\mathcal{P} \vDash [Q]$ if and only if $\mathcal{P}' \vDash [Q]$.*

Proof. Assume first that $\mathcal{P} \vDash [Q]$ and consider any runs $r'_1, \ldots, r'_n \in \mathcal{R}(\mathcal{P}')$. We will construct a run $r' \in \mathcal{R}(\mathcal{P}')$ such that $r'(q_i) = r'_i(q_i)$ for any $i \in \{1, \ldots, n\}$. Indeed, by Lemma 1, there are runs $r_1, \ldots, r_n \in \mathcal{R}(\mathcal{P})$ that match runs $r'_1, \ldots, r'_n$ on $Ch(G_X)$. By the assumption that $\mathcal{P} \vDash [Q]$, there must be a run $r \in \mathcal{R}(\mathcal{P})$ such that $r(q_i) = r_i(q_i)$ for all $i \in \{1, \ldots, n\}$. Hence, $r(q_i) = r_i(q_i) = r'_i(q_i)$ for all $i \in \{1, \ldots, n\}$. Let $r'$ be a restriction of run $r$ to the channels in $G_X$. Run $r'$ satisfies the local conditions at any party in $G_X$ because it is equal to $r$ on $Ch(G_X)$. Thus, $r' \in \mathcal{R}(\mathcal{P}')$. Finally, we notice that $r'(q_i) = r(q_i) = r'_i(q_i)$ for any $i \in \{1, \ldots, k\}$.

Next, we will assume that $\mathcal{P}' \vDash [Q]$ and consider any runs $r_1, \ldots, r_n \in \mathcal{R}(\mathcal{P})$. We will show that there is a run $r \in \mathcal{R}(\mathcal{P})$ such that $r(q_i) = r_i(q_i)$ for all $i \in \{1, \ldots, n\}$. Indeed, let $r'_1, \ldots, r'_n$ be restrictions of runs $r_1, \ldots, r_n$ to the channels in $Ch(G_X)$. Each $r'_i$ satisfies the local conditions $L'$ at any party in $G_X$ because $r'_i$ is equal to $r_i$ on each channel in $Ch(G_X)$. Thus, $r'_i \in \mathcal{R}(\mathcal{P})$ for all $i \in \{1, \ldots, n\}$. By the assumption that $\mathcal{P}' \vDash [Q]$, there is a run $r' \in \mathcal{R}(\mathcal{P}')$ such that $r'(q_i) = r'_i(q_i) = r_i(q_i)$ for all $i \in \{1, \ldots, n\}$. By Lemma 1, there is a run $r \in \mathcal{R}(\mathcal{P})$ that matches $r'$ everywhere in $Ch(G_X)$. Therefore, $r(q_i) = r'(q_i) = r_i(q_i)$ for all $i \in \{1, \ldots, n\}$. ⊠

**Lemma 3** *For any formula $\psi \in \Phi(G_X)$, $\mathcal{P} \vDash \psi$ if and only if $\mathcal{P}' \vDash \psi$.*

Proof. We use induction on the complexity of $\psi$. The base case follows from Lemma 2, and the induction step is trivial. ⊠

The statement of Theorem 7 immediately follows from Lemma 3. ⊠

# 8 Completeness

Our main result is the following completeness theorem for the *Logic of Secrets in Collaboration Networks*:

**Theorem 8** *For any graph $G$, if $\mathcal{P} \vDash \phi$ for all finite protocols $\mathcal{P}$ over $G$, then $G \vdash \phi$.*

At the core of the proof is the construction of a finite protocol. This protocol will be formed as a composition of several simpler protocols, where each of the simpler protocols is defined recursively. The base case of this recursive definition is the parity protocol defined below.

## 8.1 Parity Protocol

Let $G$ be a graph and $A$ be a subset of $Ch(G)$. We define the "parity protocol" $\mathcal{P}_A$ over $G$ as follows. The set of values of any channel $c$ in graph $G$ is a set of pairs such that

$$V(c) = \begin{cases} \{\langle b_1, b_2\rangle \mid b_1, b_2 \in \{0,1\}\} & \text{if } c \in A \\ \{\langle b, b\rangle \mid b \in \{0,1\}\} & \text{if } c \notin A \end{cases}$$

This means that under each run $r \in \mathcal{P}_A$, the value of each channel will be a pair. We identify each of the components of such a pair with one of the two ends of the channel. If channel $c$ connects party $p$ with party $q$, then by the projection $pr_p(r(c))$ we mean the component of the pair associated with $p$, and by $pr_q(r(c))$, the component associated with $q$. Now we are ready to specify the local condition

predicates $L_p$. If $c_1, \ldots c_n$ is the list of all channels incident with $p$, then $L_p$ is the statement

$$pr_p(r(c_1)) + \ldots pr_p(r(c_n)) = 0 \mod 2.$$

**Theorem 9** *$\mathcal{P}_A$ is a finite protocol.*

Proof. We need to prove existence of at least one run that satisfies all local conditions. Indeed, consider the run $r_0$ such that $r_0(c) = \langle 0, 0\rangle$ for any channel $c$. ⊠

**Definition 8** *For any run $r$, if $r(c) = \langle b_1, b_2\rangle$, then $\oplus(r(c)) = b_1 + b_2 \mod 2$.*

**Theorem 10** *For any run $r$ of the parity protocol $\mathcal{P}_A$,*

$$\sum_{c\in A} \oplus(r(c)) = 0 \mod 2.$$

Proof. Let $P$ be the set of all parties of graph $G$. If we let $Inc(p)$ mean the set of all channels incident with party $p$, then

$$\begin{aligned}\sum_{c\in A} \oplus(r(c)) &= \sum_{c\in Ch(G)} \oplus(r(c)) - \sum_{c\notin A} \oplus(r(c)) \\ &= \sum_{p\in P} \sum_{c\in Inc(p)} pr_p(r(c)) - \sum_{c\notin A} 0 \\ &= \sum_{p\in P} 0 - 0 = 0 \mod 2.\end{aligned}$$

⊠

**Definition 9** *Assume that $\pi$ is a path in the graph $G$ such that either:*

1. *$\pi = a, c_1, c_2, \ldots, c_n, b$ is a simple path, where $a, b \in A$ and $a \neq b$, or*
2. *$\pi = c_1, c_2, \ldots, c_n, c_1$ is a simple cyclic path.*

*For any run $r$ of the parity protocol $\mathcal{P}_A$ and path $\pi$ in $G$, we introduce a function called $flip(r, \pi)$ that assigns a value from $V(c)$ to each channel $c$ of the graph $G$ as follows. For any $x \in Ch(G)$, let $r(x) = \langle x_1, x_2\rangle$, and define:*

$$flip(r,\pi)(x) = \begin{cases} \langle x_1, \neg x_2\rangle & \text{if } x = a, \\ \langle \neg x_1, \neg x_2\rangle & \text{if } x \in \{c_1, \ldots, c_n\}, \\ \langle \neg x_1, x_2\rangle & \text{if } x = b, \\ \langle x_1, x_2\rangle & \text{if } x \notin \pi. \end{cases}$$

**Theorem 11** *$flip(r, \pi) \in \mathcal{R}(\mathcal{P}_A)$ for any $r \in \mathcal{P}_A$ and path $\pi$ in $G$.*

Proof. The flip operation preserves the local conditions of protocol $\mathcal{P}_A$. ⊠

**Theorem 12** *If $|A| > 1$ and graph $G$ is connected, then for any $a \in A$ and any $v \in \{0, 1\}$ there is a run $r \in \mathcal{R}(\mathcal{P}_A)$ such that $\oplus(r(a)) = v$.*

Proof. By Theorem 9, there is a run $r$ of the protocol $\mathcal{P}_A$. Suppose that $\oplus(r(a)) \neq v$. Since $|A| > 1$ and graph $G$ is connected, there is a simple path $\pi$ that connects channel $a$ with channel $b \in A$ such that $b \neq a$. Consider run $r' = flip(r, \pi)$ and notice that $\oplus(r'(a)) = v$. ⊠

**Theorem 13** *If $|A| > 1$ and graph $G$ is connected, then $\mathcal{P}_A \not\models [A]$.*

Proof. Let $A = \{a_1, \ldots, a_k\}$. Pick any values $v_1, \ldots, v_k$ such that $v_1 + \cdots + v_k = 1 \mod 2$. By Theorem 12, there are runs $r_1, \ldots, r_k \in \mathcal{R}(\mathcal{P}_A)$ such that $r_i(a_i) = v_i$ for any $i \in \{1, \ldots, k\}$. If $\mathcal{P}_A \models [A]$, then there is a run $r \in \mathcal{R}(\mathcal{P}_A)$ such that $r(a_i) = r_i(a_i)$ for any $i \in \{1, \ldots, k\}$. Therefore, $r(a_1) + \cdots + r(a_k) = r_1(a_1) + \cdots + r_k(a_k) = v_1 + \cdots + v_k = 1 \mod 2$. This contradicts Theorem 10. ⊠

**Theorem 14** *For any set $B \subseteq Ch(G)$, any $b \in B$, and any end $u$ of channel $b$, at least one of the following conditions is true:*

1. *there is a simple path that starts at party $u$, ends with a channel from $A$, and does not contain any channels from $B$,*
2. *there is a non-trivial cut $(X, Y)$ of the graph $G$ such that $A \subseteq E(X)$ and the set of crossing channels of this cut is a subset of $B$.*

Proof. Remove all channels in set $B$ from graph $G$. We will refer to this new graph as $G'$. Let $G'_u$ be the connected component of $G'$ containing party $u$. If $G'_u$ contains no channels from $A$, then this connected component defines a cut of the original graph $G$ whose only crossing channels are channels from $B$. If $G'_u$ contains at least one channel from $A$, then there is a simple path in $G'_u$ that starts at party $u$ and ends with a channel from $A$. ⊠

**Theorem 15** *For any set $B \subseteq Ch(G)$ and any $b \in B$, at least one of the following conditions is true:*

1. *there is a simple path that starts and ends with channels from $A$, contains channel $b$, and does not contain any other channels from $B$,*
2. *there is a simple cyclic path that contains channel $b$ and does not contain any other channels from $B$,*
3. *there is a non-trivial cut $(X, Y)$ of the graph $G$ such that $A \subseteq E(X)$ and the set of the crossing channels of this cut is a subset of $B$.*

Proof. Let channel $b$ connect parties $u$ and $v$. Remove all channels in set $B$ from graph $G$. We will refer to this new graph as $G'$. Let $G'_u$ and $G'_v$ be the connected components of $G'$ containing parties $u$ and $v$, respectively.

If $G'_u = G'_v$, then there is a simple path from $u$ to $v$ in $G'$. Thus, there is a cyclic path in $G$ that contains $b$ and no other channels from set $B$. (This case also includes the situation when $u = v$.)

If either $G'_u$ or $G'_v$ contains no channels from $A$, then that connected component defines a cut of the original graph $G$ whose only crossing channels are channels from $B$.

Finally, assume that $G'_u$ and $G'_v$ are two different connected components both of which contain channels from $A$. Let $a_u \in G'_u \cap A$ and $a_v \in G'_u \cap A$. Thus, the original graph $G$ contains a simple path that starts with channel $a_u$, ends with channel $a_v$, and contains channel $b$ but no other channels from set $B$. ⊠

We are ready now to prove the key property of the parity protocol that, together with Theorem 13, will be used in the next section.

**Theorem 16** *For any set $B \subseteq Ch(G)$, if there are no non-trivial cuts $(X, Y)$ of the graph $G$ such that $A \subseteq E(X)$ and the set of the crossing channels of this cut is a subset of $B$, then $\mathcal{P}_A \vDash [B]$.*

Proof. Let $B = \{b_1, \ldots, b_k\}$. Consider any runs $r_1, \ldots, r_k \in \mathcal{R}(\mathcal{P}_A)$. We will prove that there is a run $r \in \mathcal{R}(\mathcal{P}_A)$ such that $r(b_i) = r_i(b_i)$ for all $i \in \{1, \ldots, k\}$. Indeed, by Theorem 9, protocol $\mathcal{P}_A$ has at least one run $\hat{r}$. We will modify run $\hat{r}$ to satisfy conditions $\hat{r}(b_i) = r_i(b_i)$ for all $i \in \{1, \ldots, k\}$. Our modification will consist of repeating the following procedure for each $i \in \{1, \ldots, k\}$:

1. if $b_i \notin A$ and $\hat{r}(b_i) \neq r_i(b_i)$, then, by Theorem 15, there is a path $\pi$ that contains $b$ and no other element of $B$. This path is either cyclic or starts and ends with an channel from $A$. Modify $\hat{r}$ to be $flip(\hat{r}, \pi)$. This modification guarantees that $\hat{r}(b_i) = r_i(b_i)$. Since path $\pi$ does not include any channels from $B$ except $b$, the values of $\hat{r}(b_j)$ for all $j \neq i$ are not changed.

2. if $b_i \in A$ and $\hat{r}(b_i) \neq r_i(b_i)$, then, assuming that channel $b$ connects parties $u$ and $v$, at least one of the following is true: $pr_u(\hat{r}(b_i)) \neq pr_u(r_i(b_i))$ or $pr_v(\hat{r}(b_i)) \neq pr_v(r_i(b_i))$. In the first case, by Theorem 14, there is a simple path $c_1, \ldots, c_n$ that starts with party $u$, ends with a channel from set $A$, and does not contain any channels from set $B$. Let $\pi'$ be a path $b, c_1, \ldots, c_n$. Modify $\hat{r}$ to be $flip(\hat{r}, \pi')$. If $pr_v(\hat{r}(b_i)) \neq pr_v(r_i(b_i))$, make a similar modification at node $v$. These modifications guarantee that $\hat{r}(b_i) = r_i(b_i)$. Since the paths do not include any channels from $B$ except $b$, the values of $\hat{r}(b_j)$ for all $j \neq i$ are not changed.

Let $r$ be $\hat{r}$ with all the modifications described above. These modifications guarantee that $r(b_i) = \hat{r}(b_i) = r_i(b_i)$ for all $i \in \{1, \ldots, k\}$. ⊠

### 8.2 Recursive Construction

In this section we will generalize the parity protocol through a recursive construction. First, however, we will establish a technical result that we will need for this construction.

**Theorem 17 (protocol extension)** *For any cut $(X, Y)$ of graph $G$ and any finite protocol $\mathcal{P}'$ on truncation $G_X$, there is a finite protocol $\mathcal{P}$ on $G$ such that for any set $Q \subseteq Ch(G)$,*

$$\mathcal{P} \vDash [Q] \quad \textit{iff} \quad \mathcal{P}' \vDash [Q \cap E(G_X)]$$

Proof. To define protocol $\mathcal{P}$ we need to specify a set of values $V(c)$ for each channel $c \in Ch(G)$ and the set of local conditions for each party $p$ in graph $G$. If $c \in Ch(G_X)$, then let $V(c)$ be the same as in protocol $\mathcal{P}$. Otherwise, $V(c) = \{\epsilon\}$, where $\epsilon$ is an arbitrary element. The local conditions at the parties in $X$ are the same as in protocol $\mathcal{P}'$, and the local conditions at the parties in $Y$ are equal to the boolean constant $True$. This completes the definition of $\mathcal{P}$. Clearly, $\mathcal{P}$ has at least one run as long as $\mathcal{P}'$ has a run.

($\Rightarrow$) : Suppose that $Q \cap E(G_X) = \{q_1, \ldots, q_k\}$. Note that due to the soundness of the Monotonicity Axiom, $\mathcal{P} \vDash [q_1, \ldots, q_k]$. Consider any $r'_1, \ldots, r'_k \in \mathcal{R}(\mathcal{P}')$. Define runs $r_1, \ldots, r_k$ as follows:

$$r_i(c) = \begin{cases} r'_i(c) & \text{if } c \in Ch(G_X), \\ \varepsilon & \text{if } c \notin Ch(G_X). \end{cases}$$

Note that runs $r_i$ and $r'_i$, by definition, are equal on any channel incident with any party in graph $G_X$. Thus, $r_i$ satisfies the local conditions at any such party. Hence, $r_i \in \mathcal{R}(\mathcal{P})$ for any $i \in \{1, \ldots, k\}$. Since $\mathcal{P} \vDash [q_1, \ldots, q_k]$, there is a run $r \in \mathcal{R}(\mathcal{P})$ such that $r(q_i) = r_i(q_i)$ for any $i \in \{1, \ldots, k\}$. Define $r'$ to be a restriction of $r$ on subgraph $G_X$. Note that $r'$ satisfies all local conditions of $\mathcal{P}'$. Thus, $r' \in \mathcal{R}(\mathcal{P}')$. At the same time, $r'(q_i) = r_i(q_i) = r'_i(q_i)$.

($\Leftarrow$) : Suppose that $Q = \{q_1, \ldots, q_k\}$. Consider any $r_1, \ldots, r_k \in \mathcal{R}(\mathcal{P})$, and let $r'_1, \ldots, r'_k$ be their respective restrictions to subgraph $G_X$. Since, for any $i \in \{1, \ldots, k\}$, run $r'_i$ satisfies the local conditions of $\mathcal{P}'$ at any node of graph $G_X$, we can conclude that $r'_1, \ldots, r'_k \in \mathcal{R}(\mathcal{P}')$. By the assumption that $\mathcal{P}' \vDash [Q \cap E(G_X)]$, there is a run $r' \in$

$\mathcal{R}(\mathcal{P}')$ such that $r'(q) = r'_i(q)$ for any $q \in Q \cap E(G_X)$. In addition, $r'(q) = \varepsilon = r'_i(q)$ for any $q \in Q \backslash E(G_X)$. Hence, $r'(q_i) = r'_i(q_i)$ for any $i \in \{1, \ldots, k\}$. Define run $r$ as follows:

$$r(c) = \begin{cases} r'(c) & \text{if } c \in Ch(G_X), \\ \varepsilon & \text{if } c \notin Ch(G_X). \end{cases}$$

Note that $r$ satisfies the local conditions of $\mathcal{P}$ at all nodes. Thus, $r \in \mathcal{R}(\mathcal{P})$. In addition, $r(q_i) = r'(q_i) = r'_i(q_i)$ for all $i \in \{1, \ldots, k\}$. ⊠

We will now prove another key theorem in our construction. The proof of this theorem recursively defines a generalization of the parity protocol.

**Theorem 18** *For any sets $A, B_1, \ldots, B_n \subseteq Ch(G)$, if $G \nvdash \bigwedge_{1 \leq i \leq n}[B_i] \rightarrow [A]$, then there is a finite protocol $\mathcal{P}$ over $G$ such that $\mathcal{P} \vDash [B_i]$ for all $1 \leq i \leq n$ and $\mathcal{P} \nvDash [A]$.*

Proof. We use induction on the number of parties in $G$.

1. If $|A| \leq 1$, then, by the Small Set Axiom, $G \vdash [A]$. Hence, $G \vdash \bigwedge_{1 \leq i \leq n}[B_i] \rightarrow [A]$, which is a contradiction.

2. If the vertices of graph $G$ can be divided into two non-trivial disconnected sets $X$ and $Y$, then, by the Disconnected Sets Axiom,

   $$G \vdash [A \cap E(X)] \rightarrow ([A \cap E(Y)] \rightarrow [A]).$$

   Thus, taking into account the assumption that $G \nvdash \bigwedge_{1 \leq i \leq n}[B_i] \rightarrow [A]$, either

   $$G \nvdash \bigwedge_{1 \leq i \leq n} [B_i] \rightarrow [A \cap E(X)],$$

   or

   $$G \nvdash \bigwedge_{1 \leq i \leq n} [B_i] \rightarrow [A \cap E(Y)].$$

   Without loss of generality, we will assume the former. By the Monotonicity Axiom,

   $$G \nvdash \bigwedge_{1 \leq i \leq n} [B_i \cap E(X)] \rightarrow [A \cap E(X)].$$

   By the Small Set Axiom,

   $$G \nvdash [\varnothing] \rightarrow (\bigwedge_{1 \leq i \leq n} [B_i \cap E(X)] \rightarrow [A \cap E(X)]).$$

   By the Truncation Rule,

   $$G_X \nvdash \bigwedge_{1 \leq i \leq n} [B_i \cap E(X)] \rightarrow [A \cap E(X)].$$

   By the Induction Hypothesis, there is a protocol $\mathcal{P}'$ on $G_X$ such that

   $$\mathcal{P}' \vDash [B_i \cap E(X)],$$

   for any $i \in \{1, \ldots, n\}$, and

   $$\mathcal{P}' \nvDash [A \cap E(X)].$$

   Therefore, by Theorem 17, there is a protocol $\mathcal{P}$ on $G$ such that $\mathcal{P} \vDash [B_i]$, for any $i \in \{1, \ldots, n\}$, and $\mathcal{P} \nvDash [A]$.

3. Suppose there is a non-trivial cut $(X, Y)$ of graph $G$ such that $A \subseteq Ch(G_X)$ and the set of all crossing channels $C$ of this cut is a subset of $B_{i_0}$ for some $i_0 \in \{1, \ldots, n\}$. By the assumption of the theorem,

   $$G \nvdash [B_{i_0}] \rightarrow (\bigwedge_{1 \leq i \leq n} [B_i] \rightarrow [A]).$$

   By the Monotonicity Axiom,

   $$G \nvdash [C] \rightarrow (\bigwedge_{1 \leq i \leq n} [B_i \cap E(G_X)] \rightarrow [A]).$$

   By the Truncation Rule,

   $$G_X \nvdash \bigwedge_{1 \leq i \leq n} [B_i \cap E(G_X)] \rightarrow [A].$$

   By the Induction Hypothesis, there is a protocol $\mathcal{P}'$ on $G_X$ such that

   $$\mathcal{P}' \vDash [B_i \cap E(G_X)],$$

   for any $i \in \{1, \ldots, n\}$, and

   $$\mathcal{P}' \nvDash [A].$$

   Therefore, by Theorem 17, there is a protocol $\mathcal{P}$ on $G$ such that $\mathcal{P} \vDash [B_i]$, for any $i \in \{1, \ldots, n\}$, and $\mathcal{P} \nvDash [A]$.

4. Assume now that (i) $|A| > 1$, (ii) graph $G$ is connected, and (iii) for any $i \in \{1, \ldots, n\}$ there are no non-trivial cuts $(X, Y)$ of graph $G$ such that $A \subseteq Ch(G_X)$ and the set of all crossing channels of this cut is a subset of $B_i$. Consider the parity protocol $\mathcal{P}_A$ over $G$. By Theorem 13, $\mathcal{P}_A \nvDash [A]$. By Theorem 16, $\mathcal{P}_A \vDash [B_i]$ for any $i \in \{1, \ldots, n\}$.

⊠

### 8.3 Protocol Composition

**Definition 10** *For any protocols $\mathcal{P}^1 = (V^1, L^1), \ldots, \mathcal{P}^n = (V^n, L^n)$ over a graph $G$, we define the Cartesian composition $\mathcal{P}^1 \times \mathcal{P}^2 \times \cdots \times \mathcal{P}^n$ to be a pair $(V, L)$ such that*

1. $V(c) = V^1(c) \times \cdots \times V^n(c)$,

2. $L_p(\langle c_1^1, \ldots, c_1^n\rangle, \ldots, \langle c_k^1, \ldots, c_k^n\rangle)$ *if and only if* $\bigwedge_{1\le i\le n} L_p^i(c_1^i, \ldots, c_k^i)$,

For each composition $\mathcal{P} = \mathcal{P}^1 \times \mathcal{P}^2 \times \cdots \times \mathcal{P}^n$ , we let $\{r(c)\}_i$ denote the $i$th component of the value of secret $c$ over run $r$.

**Theorem 19** *For any $n > 0$ and any finite protocols $\mathcal{P}^1, \ldots, \mathcal{P}^n$ over a graph $G$, $\mathcal{P} = \mathcal{P}^1 \times \mathcal{P}^2 \times \cdots \times \mathcal{P}^n$ is a finite protocol over a graph $G$.*

Proof. We need to show that $\mathcal{P}$ has at least one run. Indeed, let $r^1, \ldots, r^n$ be runs of $\mathcal{P}^1, \ldots, \mathcal{P}^n$. Define $r(c)$ to be $\langle r^1(c), \ldots, r^n(c)\rangle$. It is easy to see that $r$ satisfies the local conditions $L_p$ for any party $p$ of the graph $G$. Thus, $r \in \mathcal{R}(\mathcal{P})$. ⊠

**Theorem 20** *For any protocol $\mathcal{P} = \mathcal{P}^1 \times \mathcal{P}^2 \times \cdots \times \mathcal{P}^n$ over a graph $G$ and any set of channels $Q$,*

$$\mathcal{P} \vDash [Q] \quad \text{if and only if} \quad \forall i\ (\mathcal{P}^i \vDash [Q]).$$

Proof. Let $Q = \{q_1, \ldots, q_\ell\}$.

($\Rightarrow$) : Assume $\mathcal{P} \vDash [Q]$ and pick any $i_0 \in \{1, \ldots, n\}$. We will show that $\mathcal{P}^{i_0} \vDash [Q]$. Pick any runs $r'_1, \ldots, r'_\ell \in \mathcal{R}(\mathcal{P}^{i_0})$. For each $i \in \{1, \ldots, i_0 - 1, i_0 + 1, \ldots, n\}$, select an arbitrary run $r^i \in \mathcal{R}(\mathcal{P}^i)$. We then define a series of composed runs $r_j$ for $j \in \{1, \ldots, \ell\}$ by

$$r_j(c) = \langle r^1(c), \ldots, r^{i_0-1}(c), r'_j(c), r^{i_0+1}(c), \ldots, r^n(c)\rangle,$$

for each secret $c \in Ch(G)$. Since the component parts of each $r_j$ belong in their respective sets $\mathcal{R}(\mathcal{P}^i)$, the composed runs are themselves members of $\mathcal{R}(\mathcal{P})$. By our assumption, $\mathcal{P} \vDash [Q]$, thus there is $r \in \mathcal{R}(\mathcal{P})$ such that $r(q_i) = r_i(q_i)$ for any $i_0 \in \{1, \ldots, \ell\}$. Finally, we consider the run $r^*$, where $r^*(c) = \{r(c)\}_{i_0}$ for each $c \in Ch(G)$. That is, we let the value of $r^*$ on $c$ be the $i_o^{th}$ component of $r(c)$. By definition of composition, $r^* \in \mathcal{R}(\mathcal{P}^{i_0})$, and it matches the original $r'_1, \ldots, r'_\ell \in \mathcal{R}(\mathcal{P}^{i_0})$ on channels $q_1, \ldots, q_\ell$, respectively. Hence, we have shown that $\mathcal{P}^{i_0} \vDash [Q]$.

($\Leftarrow$) : Assume $\forall i\ (\mathcal{P}^i \vDash [Q])$. We will show that $\mathcal{P} \vDash [Q]$. Pick any runs $r_1, \ldots, r_\ell \in \mathcal{R}(\mathcal{P})$. For each $i \in \{1, \ldots, n\}$, each $j \in \{1, \ldots, \ell\}$, and each channel $c$, let $r_j^i(c) = \{r_j(c)\}_i$. That is, for each $c$, define a run $r_j^i$ whose value on channel $c$ equals the $i$th component of $r_j(c)$. Note that by the definition of composition, for each $i$ and each $j$, $r_j^i$ is a run in $\mathcal{R}(\mathcal{P}^i)$. Next, for each $i \in \{1, \ldots, n\}$, we use the fact that $\mathcal{P}^i \vDash [Q]$ to construct a run $r^i \in \mathcal{R}(\mathcal{P}^i)$ such that $r^i(q_j) = r_j^i(q_j)$. Finally, we compose these $n$ runs $r^1, \ldots, r^n$ to get run $r \in \mathcal{R}(\mathcal{P})$. We note that the value of each channel $q_j$ on $r$ matches the the value of $q_j$ in run $r_j \in \mathcal{R}(\mathcal{P})$, demonstrating that $\mathcal{P} \vDash [Q]$. ⊠

We are now ready to prove the completeness theorem, which appeared earlier as Theorem 8:

**Theorem** *For any graph $G$, if $\mathcal{P} \vDash \phi$ for all finite protocols $\mathcal{P}$ over $G$, then $G \vdash \phi$.*

Proof. We give a proof by contradiction. Let $X$ be a maximal consistent set of formulas from $\Phi(G)$ that contains $\neg\phi$. Let $\{A_1, \ldots, A_n\} = \{A \subseteq Ch(G) \mid G \nvdash [A]\}$ and $\{B_1, \ldots, B_k\} = \{B \subseteq Ch(G) \mid G \vdash [B]\}$. Thus, $G \nvdash \bigwedge_{1\le j\le k}[B_j] \to [A_i]$, for any $i \in \{1, \ldots, n\}$. We will construct a protocol $\mathcal{P}$ such that $\mathcal{P} \nvDash [A_i]$ for any $i \in \{1, \ldots, n\}$ and $\mathcal{P} \vDash [B_j]$ for any $j \in \{1, \ldots, k\}$.

First consider the case where $n = 0$. Pick any symbol $\epsilon$ and define $\mathcal{P}$ to be $\langle V, L\rangle$ such that $V(c) = \{\epsilon\}$ for any $c \in Ch(G)$ and local condition $L_p$ to be the constant $True$ at any party. By Definition 5, $\mathcal{P} \vDash [C]$ for any $C \subseteq Ch(G)$.

We will assume now that $n > 0$. By Theorem 18, there are finite protocols $\mathcal{P}^1, \ldots, \mathcal{P}^n$ such that $\mathcal{P}^i \nvDash [A_i]$ and $\mathcal{P}^i \vDash [B_j]$ for all $j \in \{1, \ldots, k\}$. Consider the composition $\mathcal{P}$ of protocols $\mathcal{P}^1, \ldots, \mathcal{P}^n$. By Theorem 20, $\mathcal{P} \nvDash [A_i]$ for any $i \in \{1, \ldots, n\}$ and $\mathcal{P} \vDash [B_j]$ for any $j \in \{1, \ldots, j\}$.

By induction on structural complexity of any formula $\psi \in \Phi(G)$, one can show now that $G \vdash \psi$ if and only if $\psi \in X$. Thus, $\mathcal{P} \vDash \neg\phi$. Therefore, $\mathcal{P} \nvDash \phi$, which is a contradiction. ⊠

**Corollary 1** *The set $\{(G, \phi) \mid G \vdash \phi\}$ is decidable.*

Proof. The complement of this set is recursively enumerable due to the completeness of the system with respect to finite protocols. ⊠

## 9 Conclusion

We have presented a formal logical system for reasoning about an independence relation and proved the completeness of this system with respect to a semantics of secrets.

As an extension, one could study a natural generalization of this result to secrets shared by more than two parties. In that setting, a collaboration network is a hypergraph whose edges (channels) may connect arbitrary number of vertices (parties).

## References


[1] Torben Amtoft and Anindya Banerjee. A logic for information flow analysis with an application to forward slicing of simple imperative programs. *Sci. Comput. Program.*, 64(1):3–28, 2007.

[2] Ellis Cohen. Information transmission in computational systems. In *Proceedings of Sixth ACM Symposium on Operating Systems Principles*, pages 113–139. Association for Computing Machinery, 1977.

[3] Joseph A Goguen and José Meseguer. Security policies and security models. In *Proceedings of IEEE Symposium on Security and Privacy*, pages 11–20, 1982.

[4] Joseph Y. Halpern and Kevin R. O'Neill. Secrecy in multiagent systems. In *Proceedings of the Fifteenth IEEE Computer Security Foundations Workshop*, pages 32–46, 2002.

[5] Joseph Y. Halpern and Kevin R. O'Neill. Secrecy in multiagent systems. *ACM Trans. Inf. Syst. Secur.*, 12(1):1–47, 2008. (originally appeared as [4]).

[6] Donald MacKenzie. *Mechanizing Proof: Computing, Risk, and Trust*. MIT Press, 2004.

[7] Sara Miner More and Pavel Naumov. An independence relation for sets of secrets. In H. Ono, M. Kanazawa, and R. de Queiroz, editors, *Proceedings of 16th Workshop on Logic, Language, Information and Computation (Tokyo, 2009), LNAI 5514*, pages 296–304. Springer, 2009.

[8] Andrei Sabelfeld and Andrew C. Myers. Language-based information-flow security. *IEEE Journal on Selected Areas in Communications*, 21(1):5–19, 2003.

[9] David Sutherland. A model of information. In *Proceedings of Ninth National Computer Security Conference*, pages 175–183, 1986.