IEICE TRANS. COMMUN., VOL.E103–B, NO.4 APRIL 2020

PAPER

Efficient Reliability Evaluation of Multi-Domain Networks with Secure Intra-Domain Privacy

Atsushi TANIGUCHI[†,††a)], **Takeru INOUE**[††b)], **Kohei MIZUNO**[††], **Takashi KURIMOTO**[†,†††], **Atsuko TAKEFUSA**[†,†††], ***and*** **Shigeo URUSHIDANI**[†,†††], ***Members***

SUMMARY Communication networks are now an essential infrastructure of society. Many services are constructed across multiple network domains. Therefore, the reliability of multi-domain networks should be evaluated to assess the sustainability of our society, but there is no known method for evaluating it. One reason is the high computation complexity; i.e., network reliability evaluation is known to be #P-complete, which has prevented the reliability evaluation of multi-domain networks. The other reason is intra-domain privacy; i.e., network providers never disclose the internal data required for reliability evaluation. This paper proposes a novel method that computes the lower and upper *bounds* of reliability in a distributed manner without requiring privacy disclosure. Our method is solidly based on graph theory, and is supported by a simple protocol that secures intra-domain privacy. Experiments on real datasets show that our method can successfully compute the reliability for 14-domain networks in one second. The reliability is bounded with reasonable errors; e.g., bound gaps are less than 0.1% for reliable networks.

key words: *network reliability, social sustainability, inter-domain networks, graph theory*

1. Introduction

Social sustainability now largely depends on communication networks, as they are an essential infrastructure of contemporary society. To assess the sustainability of communication networks, their *reliability* should be known when designing network services. Network reliability [1]–[4] is defined in terms of connectivity between given terminals, i.e., the probability of connecting the terminals on a probabilistic network where link failure mirrors a stochastic process. Since connectivity is a necessary condition for network services to work, network reliability is considered to be a fundamental metric of communication networks [1]–[4]. The research community is devoting a lot of effort to develop reliability evaluation methods that can assess the network tolerance to natural disasters [5], [6].

This paper studies network reliability for *multi*-domain networks. Today's communication networks consist of multiple providers or domains. Network services are often deployed across multiple domains as end-to-end service delivery encompasses several regions and business continuity must be assured even under severe accidents. There have been many efforts directed towards constructing network services across multiple domains, e.g., cost minimization algorithms [7]–[9] and standardized protocols [10], [11]. However, to the best of our knowledge, no paper has studied network reliability in the context of multi-domain networks.

This paper considers two types of players: domain providers (DPs) and service providers (SPs). DPs manage their own domains and evaluate the reliability of their own domains. A SP provides a network service across the domains and so must compute reliability as a whole. Figure 1(a) shows an instance of network reliability evaluation for three domains. The domains are connected via border nodes, and every domain has terminals. Every link could fail with some specified probability, e.g., 1%. Our problem is to compute the probability of achieving terminal connection; that is 96.9817% in this instance.

The problem seems identical with the traditional reliability evaluation assuming a single-domain network, but the existence of multi domains introduces the following challenges.

- Computation complexity. Reliability evaluation is known to be #P-complete [12], [13]. #P is the complexity class of enumeration problems associated with NP decision problems, and a #P problem must be at least as hard as the corresponding NP problem (for network reliability, the corresponding NP problem is to find *any* single network state connecting the terminals, while the #P problem is to assess *all* connected states). In to face of this computational hardness, the recent work of [14]–[16] succeeded in computing the reliability for networks with less than 200 links. Unfortunately, multi-domain networks must be larger than single-domain ones, so the computation issue is more challenging. To reduce the computation burden, sampling approaches like Monte Carlo simulations have been studied [4]. This approach, however, provides no guarantee as to the accuracy and could result in large errors [15], [17]. This implies that we might overlook the significant risk of network unreliability, which would cause terrible disruption in the future.
- Intra-domain privacy. DPs remain reluctant to disclose their internal information, e.g., the network topology

Manuscript received May 23, 2019.
Manuscript revised August 5, 2019.
Manuscript publicized September 27, 2019.
[†]The authors are with Graduate University for Advanced Studies (SOKENDAI), Kanagawa, 240-0193 Japan.
[††]The authors are with NTT Network Innovation Laboratories, NTT Corporation, Yokosuka-shi, 239-0847 Japan.
[†††]The authors are with National Institute of Informatics, Tokyo, 101-8430 Japan.
a) E-mail: atsushi.taniguchi.hn@hco.ntt.co.jp
b) E-mail: takeru.inoue.dr@hco.ntt.co.jp
DOI: 10.1587/transcom.2019EBP3119

and the link availabilities, because such disclose might allow their competitors to estimate business strategies and allow attackers to find their vulnerabilities. No work has, to our knowledge, studied this issue in the context of network reliability. Reference [8], [9] proposed an cost minimization method for multi-domain networks. This method utilizes secure multi-party computation to keep the internal information private. This method was designed for NP problems that can be efficiently solved by pruning the search space, but it cannot be applied to our #P problem, as we have to examine the whole search space in unitary manner.

This paper proposes a novel method that efficiently computes the reliability of multi-domain networks without revealing intra-domain privacy. Our method allows us to partition the problem so as to yield upper and lower bounds of reliability. Each DP computes the reliability of their domain, and the SP then unifies the results to yield the bounds for the whole network. Our contributions are summarized as follows.

- Theory: This paper develops a rigorous theory for an effective partition. The partition reduces the problem size to decrease computation complexity. In addition, the partition guarantees that no intra-domain information is disclosed. The theory utilizes the graph contraction technique to yield upper and lower bounds of reliability. It is worth noting that the bounds of our method have clear advantage against the sampling approach that has no error bounds [4]; if the network were unreliable, we are assured of realizing this by the *small* lower bound; if the lower bound is high, it means that the network is assured of being reliable enough.
- Protocol: This paper defines a primitive protocol between the SP and DPs. DPs can compute their domains' reliability without revealing their internal data. The computed reliabilities are processed by the SP using secure computation techniques. Several practical issues including inter-domain connections are also addressed.
- Experiments: Our method is numerically evaluated using several real networks. While the recent work of [14]–[16] could deal with only networks having fewer than 200 links in total, our method is shown to successfully evaluate the reliability of 14 domains with 907 links. The bound gaps are reasonably small.

The rest of this paper is organized as follows. Section 2 formalizes the problem addressed. Section 3 establishes the theory, while Sect. 4 describes the protocol. Section 5 reports our experiments and their results. Sections 6 and 7 discuss related work and our conclusions, respectively.

2. Problem Statement

This section provides the problem statements needed for understanding our advances. Section 2.1 defines our network model, and Sect. 2.2 describes the problem raised by reliability evaluations of multi-domain networks.

2.1 Network Model

This paper does not focus on any specific type of network. Networks can be physical, logical, or any mixture, as long as they can be represented as our model described below. A network is represented as undirected graph $G = (V, E)$, where V is a set of nodes and E is a set of links. The whole network is partitioned into *domains*, and the domains are numbered; the set of domain numbers is denoted by $D = \{1..|D|\}$. Node set V is partitioned following the domains; i.e.,

$$\bigcup_{i \in D} V_i = V,$$
$$V_i \cap V_j = \emptyset \qquad i, j \in D\ (i \neq j).$$

Domain i is defined as the induced subgraph, $G[V_i]$.

Nodes connecting to another domain are called *border* nodes, and the set of border nodes is defined as $B \subset V$. Since every border node belongs to a single domain (from the partition definition), we can consider a function $f_B : B \to D$ and f_B is surjective (i.e., every domain has at least one border node). The set of domain i's border nodes, i.e., $B \cap V_i$, is a vertex separator[†] for the domain and the others.

The service provided by the SP consists of nodes named *terminals*. The terminal set is defined as $T \subset V$. In this paper, we assume that every domain has at least one terminal, so the surjective function $f_T : T \to D$ is considered. Without loss of generality, we assume that every terminal is not a border node; i.e., $T \cap B = \emptyset$ (if not, we can cleave the border terminal into the border-only node and a new terminal, and then connect them with a perfect link; the new terminal is not connected to the neighbors of the border node).

Given network G, let $m = |E(G)|$. The m-dimensional binary vector $\boldsymbol{x} = \{x_1, \ldots, x_m\} \in \{0, 1\}^m$ is used to represent the current status of the links; if $x_i = 0$, then link $e_i \in E$ has failed; otherwise, e_i is available. We assume that every link e_i independently fails with probability $1 - p_i$, where $p_i \in [0, 1]$ is the probability that e_i is available. Nodes are regarded as perfect. Given status $\boldsymbol{x}$, the corresponding subgraph, $G(\boldsymbol{x}) \subseteq G$, is defined by $V(G(\boldsymbol{x})) = V$ and $E(G(\boldsymbol{x})) = \{e_i \in E : x_i = 1\}$.

Network reliability is defined as follows. Given network G with T, the set, $\mathcal{G}(G, T)$, of subgraphs connecting the terminals is,

$$\mathcal{G}(G, T) = \{G(\boldsymbol{x}) \subseteq G : G(\boldsymbol{x}) \text{ connects } T\}.$$

Note that we allow detour paths, which connect terminals in the same domain via another domain (this issue is discussed in Sect. 4.1). Network reliability $R(G, T)$ can be considered

[†]A subset, $S \subset V$, of nodes is a *vertex separator* for nonadjacent nodes $u, v \in V$, if the removal of S from the graph separates u and v into distinct connected components.

as the total probability of connecting the terminals,

$$R(G,T) = \sum_{G(\boldsymbol{x})\in\mathcal{G}(G,T)} \prod_{i\in\{1..m\}} [x_i p_i + (1-x_i)(1-p_i)], \tag{1}$$

where the product term is the probability that the network is in $G(\boldsymbol{x})$.

We assume that DP i knows $G[V_i]$. We also assume that the SP figures out how to connect the domains, i.e., $G[B]$ (or the contracted graph of $G[B]$, as is discussed in Sect. 3.2). Note that inter-domain connections are often very complicated to grasp even if we limit ourselves to those used by the service, so we address this concern in Sect. 4.1.

2.2 Reliability Evaluation for Multi-Domain Networks

Our problem is defined as follows.

Problem. *SP efficiently computes $R(G,T)$, under the information constraint, i.e., $G[V_i]$ is known only to DP i while $G[B]$ is known only to the SP.*

3. Theory

This section establishes a theory that yields lower and upper bounds of reliability. The problem is partitioned to reduce computation complexity and also to secure intra-domain privacy. The theory is developed in two steps as follows (Fig. 1).

(a) Section 3.1: Compute the exact value of $R(G,T)$ when f_B is bijective; i.e., every domain has just a single border node, $\forall i \in \{1..|D|\}, |B \cap V_i| = 1$.

(b) Section 3.2: Compute $R(G,T)$ with the bounds when f_B is surjective; this is the general case.

3.1 Single Border Node

Lemma 1. *Every subgraph connecting the terminals also connects all the border nodes.*

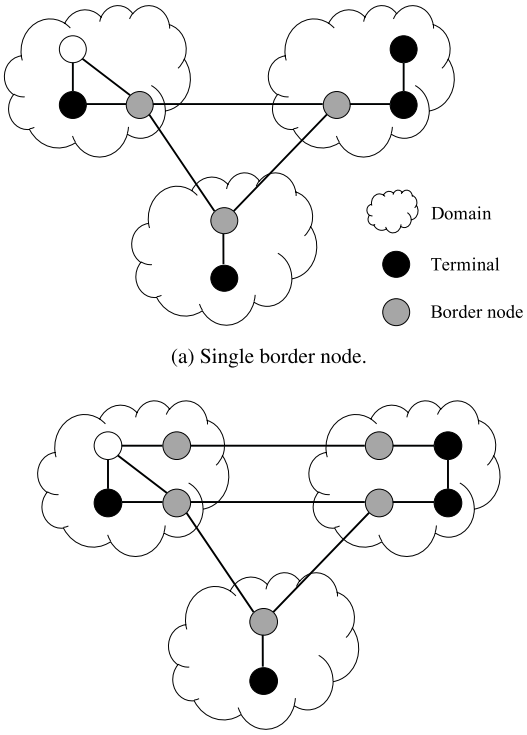
$$\mathcal{G}(G,T) = \mathcal{G}(G, T\cup B).$$

Proof. Since the right side of the equation connects T, we have $\mathcal{G}(G,T) \supseteq \mathcal{G}(G,T\cup B)$.

We then prove the converse, $\mathcal{G}(G,T) \subseteq \mathcal{G}(G,T\cup B)$, by contradiction. Assume that there exists a border node $b \in B$ that is disconnected from some of T in a subgraph of $\mathcal{G}(G,T)$. Without loss of generality, we assume the border node is in domain i, i.e., $b \in V_i$. Since b is the single border node in the domain (f_B is bijection in Problem 1), $\{b\}$ is a vertex separator for the domain, which implies that some terminals are disconnected from/to the domain. This contradicts the fact that the subgraph is in $\mathcal{G}(G,T)$. □

Corollary 1. *From* (1) *and Lemma 1, we have,*

$$R(G,T) = R(G, T\cup B).$$



(a) Single border node.

(b) General case.

Fig. 1 Problem instances.

Lemma 2. *The link set is partitioned into the domains and the backbone.*

$$E(G[B]) \cup \bigcup_{i\in D} E(G[V_i]) = E, \tag{2}$$

$$E(G[V_i]) \cap E(G[V_j]) = \emptyset \qquad i,j \in D\ (i \neq j), \tag{3}$$

$$E(G[V_i]) \cap E(G[B]) = \emptyset \qquad i \in D. \tag{4}$$

Proof. From the definition of our network model, for each link $e = \{u,v\}$, the ends are either in a domain ($u,v \in V_i$) or are borders ($u,v \in B$). □

We define the *join* operation over two sets of subgraphs, following family algebra [18]. Given two sets of subgraphs, $\mathcal{G}_1 = \mathcal{G}(G_1,T_1)$ and $\mathcal{G}_2 = \mathcal{G}(G_2,T_2)$, their join is defined, as follows,

$$\mathcal{G}_1 \sqcup \mathcal{G}_2 = \Big\{(V(G_1)\cup V(G_2), E(G_1')\cup E(G_2')) : G_1' \in \mathcal{G}_1, G_2' \in \mathcal{G}_2\Big\}.$$

Lemma 3. *The set of subgraphs connecting the terminals is given as the join between the domains and the backbone.*

$$\mathcal{G}(G, T\cup B) = \mathcal{G}(G[B], B) \sqcup \bigsqcup_{i\in D} \mathcal{G}(G[V_i], (T\cup B)\cap V_i).$$

Proof. We first prove $\mathcal{G}(G,T\cup B) \supseteq \mathcal{G}(G[B],B) \sqcup \bigsqcup_{i\in D} \mathcal{G}(G[V_i],(T\cup B)\cap V_i)$. The first term of the right side, $\mathcal{G}(G[B],B)$, indicates that all the border nodes are connected

in every subgraph. The second term, $\mathcal{G}(G[V_i], (T \cup B) \cap V_i)$, indicates that all the terminals in domain i are connected to the border node in every subgraph. Hence, every joined subgraph in the right side connects all the terminal and the border nodes, which implies the left side.

We then prove the converse: $\mathcal{G}(G, T \cup B) \subseteq \mathcal{G}(G[B], B) \sqcup \bigsqcup_{i \in D} \mathcal{G}(G[V_i], (T \cup B) \cap V_i)$. In each subgraph of the left side set, every border node is a distinct node separator for a singleton. Cutting the subgraph at the border nodes (without removing them), each piece connects either the border nodes or the border node with terminals in the domain. Note that G is covered by the union of $G[B]$ and $G[V_i]$'s in the right side from (2). The right side is, therefore, implied. □

Lemma 4. *Network reliability is partitioned as follows,*

$$R(G, T \cup B) = R(G[B], B) \prod_{i \in D} R(G[V_i], (T \cup B) \cap V_i).$$

Proof. In Lemma 2, each subgraph in the left side, $G' \in \mathcal{G}(G, T \cup B)$, is cut into the domains and the backbone in the right side. From (3) and (4), no link is shared between the domains and the backbone. The reliability of the whole network is, therefore, simply given as the product of reliabilities for the domains and for the backbone. □

Theorem 1. *Reliability of multi-domain networks can be partitioned into those of the domains and the backbone.*

$$R(G, T) = R(G[B], B) \prod_{i \in D} R(G[V_i], (T \cup B) \cap V_i).$$

Proof. From Corollary 1 and Lemma 4. □

3.2 General Case

Lemma 5.

$$\mathcal{G}(G, T) \supseteq \mathcal{G}(G, T \cup B).$$

Proof. Same as the former half of Lemma 1. □

Lemma 5 implies that some border nodes can be bypassed in (b), unlike Lemma 1.

Let $G' = (V', E')$ be the graph where border nodes in the same domain are *contracted* (Fig. 2). Contraction of a pair of nodes produces a new graph in which the two nodes are merged; their links are left as they are (some of them could be parallel links). Let $B' \subset V'$ be the set of new border nodes after the contraction.

Corollary 2. *Associating a contracted node with any of original nodes, there is the injection, f_V, between the new and original vertex sets,*

$$f_V : V' \to V.$$

Corollary 3. *There is the bijection, f_E, between the new and original link sets,*

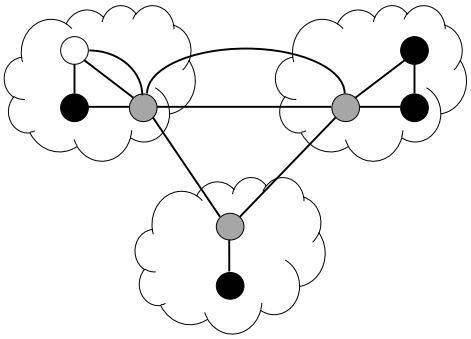


Fig. 2 Contraction of border nodes. This graph is the contracted graph, G', of Fig. 1(b). The three border nodes form the set, B'.

$$f_E : E(G'[B']) \cup \bigcup_{i \in D} E(G'[V_i']) \to E.$$

Based on Corollaries 2 and 3, nodes and links in a contracted graph are associated with those in the original graph, if needed; i.e., new links are associated with the availability of the original ones.

Lemma 6. *The set of connected subgraphs is a superset of the join between the domains and the contracted backbone.*

$$\mathcal{G}(G, T \cup B) \supseteq \mathcal{G}(G'[B'], B') \sqcup \bigsqcup_{i \in D} \mathcal{G}(G[V_i], (T \cup B) \cap V_i).$$

Proof. The second term of the right side, $\mathcal{G}(G[V_i], (T \cup B) \cap V_i)$, indicates that in domain i all the border nodes are connected. In this case, it is sufficient that the backbone connects one of border nodes for each domain; i.e., from Corollary 2, it is sufficient that all the contracted border nodes are connected, which is the first term, $\mathcal{G}(G'[B'], B')$, in the right side. □

Theorem 2. *The lower bound of the reliability is given as,*

$$R(G, T) \geq R(G'[B'], B') \prod_{i \in D} R(G[V_i], (T \cup B) \cap V_i). \tag{5}$$

Proof. From Lemma 5 and Lemma 6. □

Lemma 7.

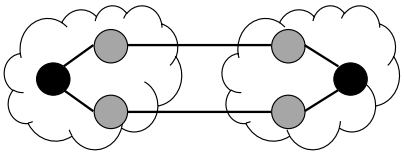
$$\mathcal{G}(G, T) \subseteq \mathcal{G}(G', T).$$

Proof. From Corollary 2, if there is a path in G, there also is a path in G'. □

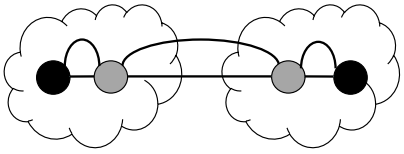
Lemma 8. *In the contracted network, the set of connected subgraphs is a subset of the join between the domains and the backbone.*

$$\mathcal{G}(G', T) = \mathcal{G}(G'[B'], B') \sqcup \bigsqcup_{i \in D} \mathcal{G}(G'[V_i'], (T \cup B') \cap V_i').$$

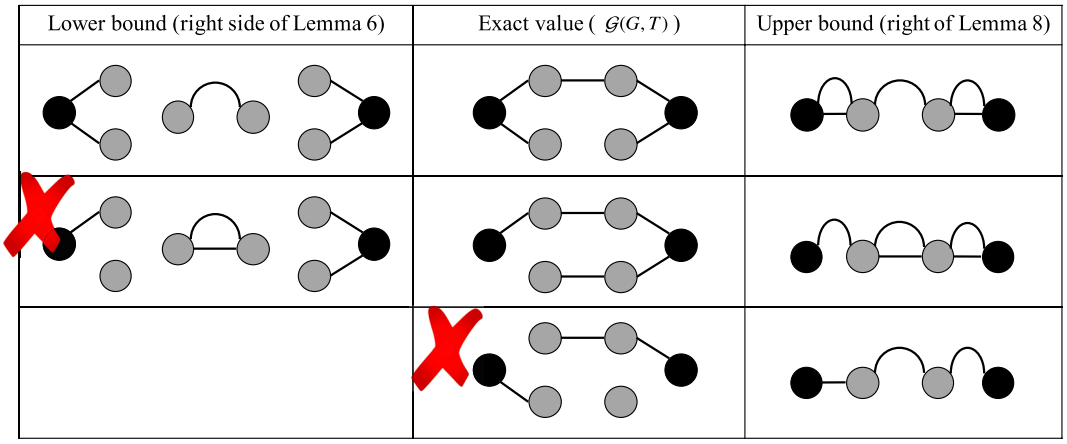
Proof. Since the contracted graph, G', has a single border node in every domain, we have an identical lemma, $\mathcal{G}(G', T) = \mathcal{G}(G', T \cup B')$, with Lemma 1. Replacing



(a) A network.



(b) Contracted graph.



(c) Some subgraphs included in the sets of lower bound, exact value, and upper bound.

Fig. 3 Subgraphs used to describe how the bounds deviate from the exact value.

$\mathcal{G}(G',T)$ with $\mathcal{G}(G',T \cup B')$ in Lemma 8, which can be proved in the same way of Lemma 3. □

Theorem 3. *The upper bound of the reliability is given as,*

$$R(G,T) \leq R(G'[B'],B') \prod_{i \in D} R(G'[V'_i],(T \cup B') \cap V'_i). \tag{6}$$

Proof. From Lemma 7 and Lemma 8. □

3.3 Examples

Figure 3 illustrates how the lower and upper bounds deviate from the exact value. Figure 3(a) is the network considered, and Fig. 3(b) gives the corresponding contracted graph. Figure 3(c) shows some subgraphs included in the set of connected subgraphs for the lower bound (i.e., the right side of Lemma 6), the set for the exact value ($\mathcal{G}(G,T)$), and the set for the upper bound (the right side of Lemma 8).

- Exact value (the center column). The top two subgraphs are connected, while the bottom one is disconnected (X-mark indicates the subgraph is not included in the set of connected subgraphs). Therefore, the top two are included in $\mathcal{G}(G,T)$, and the bottom one is not.
- Upper bound (the right column). The subgraphs shown are derived from the center column of the same row. The three subgraphs seem all connected, but the bottom one is actually not, as shown in the center column; this false positive leads to overestimation. Since no false negative happens as discussed in our theory, it can be used in determining the upper bound.
  The following observation allows us to expect tight upper bounds. Since false positive subgraphs are actually disconnected, they are likely to have many failed links. If link availabilities are small, the probabilities of these subgraphs is expected to be very small; given link availabilities of 99%, i.e., $p_i = 0.99$, the probability that the network is in the bottom state of Fig. 3(c) is $0.99^3 \times 0.01^3 = 0.00000097029$. Therefore, false positive subgraphs do not impose significant errors on the upper bound.
- Lower bound (the left column). The subgraphs are separated according to the domains, because inter-domain graphs are contracted, while intra-domain graphs are not; in each piece of each subgraph, terminals and border nodes should be connected. Although only the top subgraphs seem connected, the middle one is actually connected as shown in the center column; this false negative leads to underestimation, and it can be used

for the lower bound.

4. Practice

This section addresses the practical issues raised when solutions are needed for actual deployment. After discussing inter-domain connections in Sect. 4.1, Sect. 4.2 defines a primitive protocol between an SP and DPs to compute the reliability bounds of Theorems 2 and 3. This section aims at showing that a basic protocol can be defined for our theory; further elaboration for specific services will be done in the future.

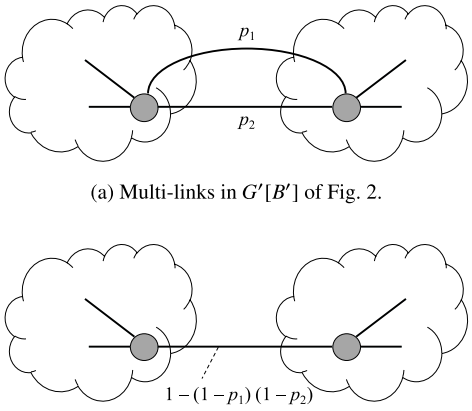
4.1 Inter-Domain Connections

In our network model, we assume that the SP can utilize the contracted graph of inter-domain network, $G'[B']$, which is included in (5) and (6). We first discuss the determination process of $G'[B']$, for (I) standardized specifications like ETSI NFV [19], [20] and (II) the general case.

(I) The ETSI NFV specifications allow SPs to retrieve adjacency between domains that join the NFV infrastructure [19]. We, therefore, assume that the SP in our method can determine the topology of $G'[B']$ based on this adjacency.

(II) In the general case, we consider *logical connections* between domains; i.e., a logical connection could be a sequence of physical links if the domains are not adjacent. This is because, in reliability evaluation, we do not need to recognize how nodes are connected; it is sufficient to know the probability that two nodes can communicate. The SP, hence, assumes that an inter-domain connection exists in $G'[B']$ if terminals in the two domains would directly exchange messages in the SP's service.

Next, we discuss the availability estimation for inter-domain connections (this discussion is applicable for (I) and (II)). In our protocol, the inter-domain availabilities are estimated by DPs and are given to the SP, as will be shown in Sect. 4.2. Although there could be multiple inter-domain connections between domains as shown in Fig. 1b, these links are not necessarily distinguished if the contracted graph, $G'[B']$, is considered. This is because the set, M, of multi-links is equivalent in reliability evaluations to a single one with availability of $1 - \prod_{i \in M}(1 - p_i)$, as shown in Fig. 4. Therefore, it is sufficient for DPs to estimate the probability that the two domains can communicate. This inter-domain availability could be estimated as follows: periodically in advance, domains continue to exchange active probes between their border nodes, so as to use the success probability as the availability; or, upon receiving a request for the availability, domains examine their history of BGP updates to compute how often their counterparts were seen through BGP, since BGP messages represent the communicability between domains [21].



(a) Multi-links in $G'[B']$ of Fig. 2.

(b) Single link.

Fig. 4 (a) Multi-links between a pair of border nodes. (b) Corresponding connection equivalent to (a) in terms of availability.

In our model, even if terminals in the same domain have no path within the domain, they are allowed to be connected via a path that detours outside the domain. Although these detoured paths are forbidden by BGP, they could be utilized if overlay networks were established between terminals of different domains.

4.2 Protocol

This subsection defines a primitive protocol that computes the reliability bounds in a distributed manner. The protocol is defined for (b). The protocol also runs for (a); in this case, the contracted graph is identical with the original one, so the lower and upper bounds match.

The protocol assumes the following initial states.

- SP: the number of terminals to be placed in domain i is fixed; a secure channel is established with DP i.
- DP i: $G[V_i]$ is fixed.

Figure 5 illustrates the protocol sequence. First, the SP notifies DP i of the number of terminals and of domains accessed from domain i. DP i then determines the nodes hosting the terminals are placed ($T \cap V_i$ has fixed). DP i finds the border nodes to the other domains ($B \cap V_i$ has fixed), and estimates availabilities for the inter-domain connections; the availabilities are sent to the SP ($G'[B']$ is fixed).

Finally, SP computes $R(G'[B'], B')$, while DP i computes $R(G[V_i], (T \cup B) \cap V_i)$ and $R(G'[V'_i], (T \cup B') \cap V'_i)$; these partial reliabilities are secretly multiplied using secure computation techniques, such as secure multi-party computation (MPC) [22] and homomorphic encryption [23] (these secure computation techniques allow computation while keeping the inputs private). In this way, no intra-domain information is disclosed in the protocol.

We describe an example of the computation procedure assuming the use of MPC. The computation is performed by *participants*, i.e., the SP and DPs in our protocol; the SP is called 0-th participant, while DP j is called j-th participant.

Every partial reliability is divided into *shares* based on cryptographic theory. Each participant is allocated a share of partial reliability, but the partial reliability can be reconstructed only when a sufficient number of shares are combined; individual shares are of no use on their own. We assume that the participants do not collude with each other. In this paper, a share of reliability R allocated to participant j is denoted by $[[R]]_j$. In the ordinary use of MPC, arithmetic operations are performed over shares, and only the result is reconstructed.

We discuss only the lower bound using Fig. 6, as the upper bound can be computed in a similar fashion. For readability, the lower bound (i.e., the right side of (5)) is denoted by R^L. The partial reliability of an inter-domain network is denoted by $R_0^L = R(G'[B'], B')$, while the partial reliability of domain i is denoted by $R_i^L = R(G[V_i], (T \cup B) \cap V_i)$. The lower bound is then written as $R^L = \prod_{i\in\{0\}\cup D} R_i^L$. Since summation is more efficient than multiplication in MPC [8], the multiplication is converted to a summation by taking the logarithm of reliabilities, i.e., $\log R^L = \sum_{i\in\{0\}\cup D} \log R_i^L$. The participants generate the shares for their partial reliability, as follows,

$$\begin{aligned}&\textsc{MpcDivide}(\log R_i^L)\\ =&\{[[\log R_i^L]]_0, [[\log R_i^L]]_1, \ldots, [[\log R_i^L]]_{|D|}\}.\end{aligned}$$

The shares with subscript j are gathered by participant j, who executes MPC summation over the shares,

$$\begin{aligned}&\textsc{MpcSum}([[\log R_0^L]]_j, [[\log R_1^L]]_j, \ldots, [[\log R_{|D|}^L]]_j)\\ =&[[\log R^L]]_j.\end{aligned}$$

The SP gathers the shares of the lower bound and reconstructs it, as follows,

$$\begin{aligned}&\textsc{MpcReconst}([[\log R^L]]_0, [[\log R^L]]_1, \ldots, [[\log R^L]]_{|D|})\\ =&\log R^L.\end{aligned}$$

The protocol overhead is briefly discussed assuming the use of MPC. Reference [8] states that the primary overhead of MPC is transmission, not computation, since transmission is slower than arithmetic operations by an order of magnitude. By taking the logarithm of partial reliabilities,

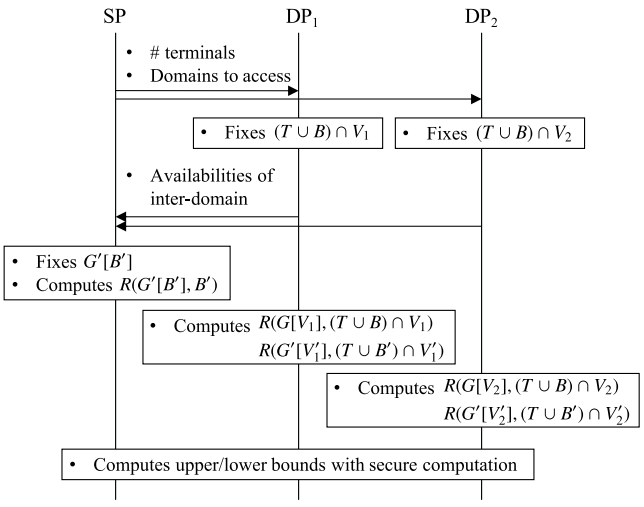


Fig. 5 A protocol between the SP and two DPs, which computes the reliability bounds. The last step is exemplified in Fig. 6.

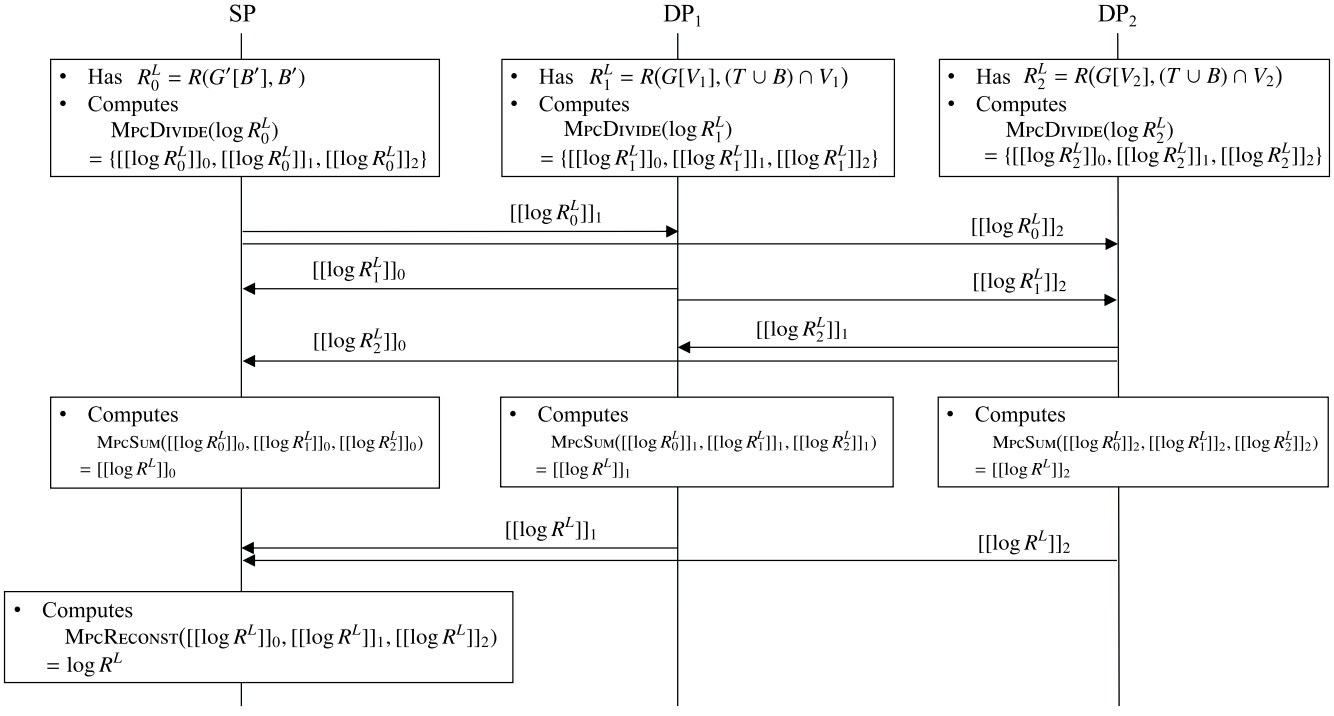


Fig. 6 An example of MPC for the lower bound in our protocol.

we can convert the multiplication in (5) and (6) into a summation, as described above. Summation requires just two parallel transmissions, i.e., distribution of partial reliabilities and collection of results in MPC. In total, our protocol has just four stages of parallel transmission: SP's notification, DPs' replies, and two transmissions for MPC, as shown in Figs. 5 and 6.

Further elaboration of the protocol, e.g., authentication and key exchange, is left as future work, because it depends on service details.

5. Experiments

This section uses real datasets to assess our method in terms of computation costs (Sect. 5.1) and bound gaps (Sect. 5.2). Since the protocol overhead is not significant as discussed in Sect. 4.2, it is not measured.

To the best of our knowledge, there is no method that can evaluate reliability while securing intra-domain privacy, so there is no direct benchmark. As a baseline, we use an existing reliability evaluation method [14]–[16], that discloses domain internal data in computing the exact reliability of the whole network. Since the existing method computes the exact value, it is used to assess the reliability bounds of our method.

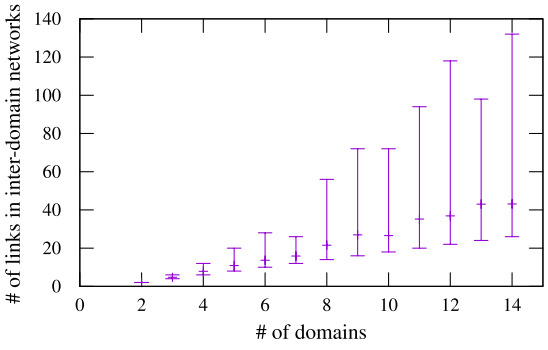
Domains are randomly chosen from the real networks in Table 1 [24]. Each domain has one or two terminals and two or four border nodes; terminals and border nodes are randomly chosen. Domains are connected assuming active-active configuration of border nodes (e.g., the upper domains in Fig. 1b). Inter-domain topologies are sampled from an AS-level network[†]; we randomly choose a starting domain (AS), from which we visit the specified number of domains in the breadth first order, then we reconstruct every link between the visited domains if existed in the original network. The number of domains ranges from 2 to 14. For each number of domains, 30 topologies are generated. For each topology, an inter-domain topology is sampled, as described above; as a result, every domain pair has a link with probability of 31.7% in our experiments. The maximum topology includes 907 links with 14 domains. Inter-domain and whole topologies, i.e., $G[B]$ and G, respectively, are summarized in Fig. 7; for each number of domains, the average is represented by marks, and the minimum and maximum are indicated by the line whiskers. Link availabilities are uniformly and randomly determined. Parameters are summarized in Table 2. Each problem instance is specified for a pair of topology and availability range.

The existing reliability evaluation method [14]–[16], which is also used in our method to compute partial reliabilities, is implemented in C++ using the internal library of [25][††]. Graph manipulation including contraction is implemented in Python. Computation was conducted on a single core of a Core i7-8550U 1.8 GHz with 5 GB RAM.

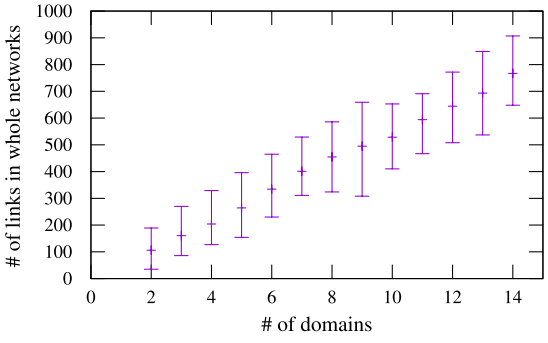
[†]http://irl.cs.ucla.edu/topology/
[††]http://graphillion.org/

Table 1 Statistics of real networks used as intra-domains.

Network	$\|V\|$	$\|E\|$
Oxford	20	26
Funet	26	30
Darkstrand	28	31
Sunet	26	32
Shentel	28	35
Bren	37	38
NetworkUsa	35	39
IowaStatewideFiberMap	33	41
PionierL1	36	41
LambdaNet	42	46
Intranetwork	39	51
RoedunetFibre	48	52
Ntelos	47	58
Palmetto	45	64
UsSignal	61	78
Missouri	67	83
Switch	74	92
VtlWavenet2008	88	92
RedBestel	84	93
Intellifiber	73	95
VtlWavenet2011	92	96
Oteglobe	83	99



(a) # of links in inter-domain networks.



(b) # of links in whole networks.

Fig. 7 The numbers of links (a) in inter-domain networks $G[B]$, and (b) in whole networks G.

5.1 Computation Costs

This subsection evaluates our method and the existing method in terms of computation time and memory usage.

Table 2 Parameter ranges.

# of terminals per domain	{1, 2}
# of border nodes per domain	{2, 4}
# of domains	{2, 3, ..., 14}
Link availability	(0.99,1), (0.999,1), or (0.9999,1)

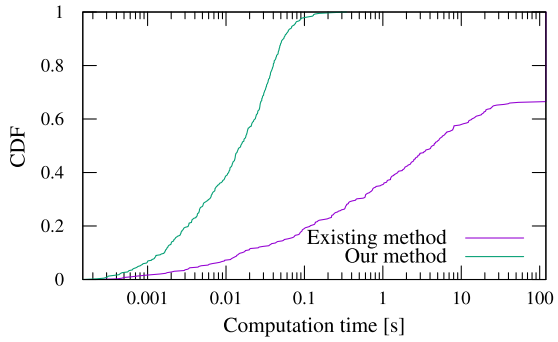


Fig. 8 Cumulative distribution function (CDF) of computation time.

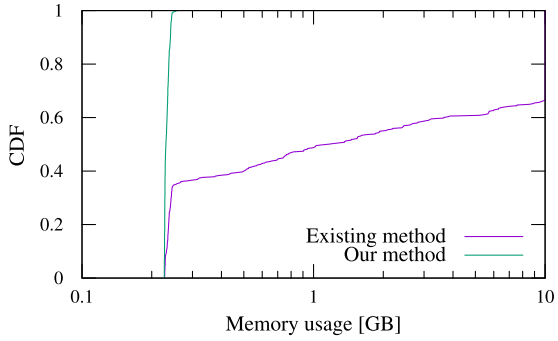


Fig. 9 Cumulative distribution function (CDF) of memory usage.

Since our method runs in parallel for the SP and DPs, we take the *maximum* of computation time and memory usage. The existing method has to run as a single process, because no distributed algorithm has been found for it. Computation was executed with a time limit of 120 [s].

Figure 8 shows the cumulative distribution function (CDF) of computation time. Our method consistently lies to the left of the existing method, which implies that our method is more efficient thanks to our partition theory. Our method solved all the instances, while the existing method only solved 66% of them. Figure 9 shows the CDF of memory usage. Our method required around less than 255 MB of memory to solve all the instances (the small deviation shown in our method means that our method requires a much smaller amount of memory compared to the amount that the OS process allocated by default). On the contrary, the existing method cannot complete even with 5 GB of memory.

Figure 10 plots the computation time against the number of domains, while Fig. 11 demonstrates similar results for memory usage. For each number of domains, the average is represented by marks, and the minimum and maximum are indicated by the line whiskers. Our method scales

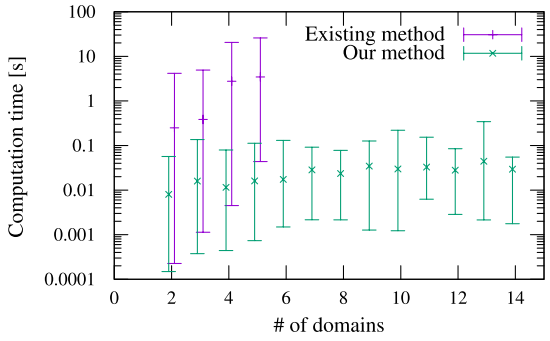


Fig. 10 Computation time versus the number of domains.

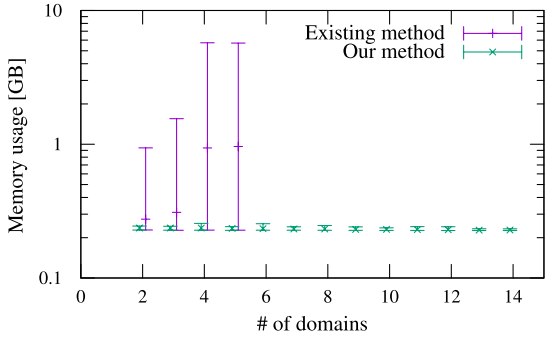


Fig. 11 Memory usage versus the number of domains.

very well, while the existing method scales poorly; the existing method could not solve some instances for six or more domains. This is because the existing method incurs exponential growth in the amount of time and memory against the number of domains, due to the nature of #P problems.

It is worth noting that even if the results of our method were multiplied by the number of domains, our method would still outperform the existing method for large domain numbers.

5.2 Bound Gaps

Figure 12 shows the gap between lower and upper bounds of our method. For each number of domains, the average is represented by marks, and the minimum and maximum are indicated by the line whiskers. The average gaps are less than 0.1 for link availabilities in (0.99,1), and they are less than 0.001 for those in (0.9999,1). The gaps grow slightly with large domain numbers, but the growth is slow.

We examine lower and upper bounds separately in Fig. 13. The figure shows lower and upper bounds for link availabilities in (0.9999,1), which are plotted against the exact reliability computed by the existing method; we had similar results for other availability ranges. Points indicate lower and upper bounds for each instance; points below the line of $y = x$ correspond to the lower bounds, while these above the line are the upper bounds. The lower bounds have larger errors than the upper bounds. This is because the right side of Lemma 6 places a strong restriction on intra-domain reliability, i.e., all of the border nodes have to be connected,

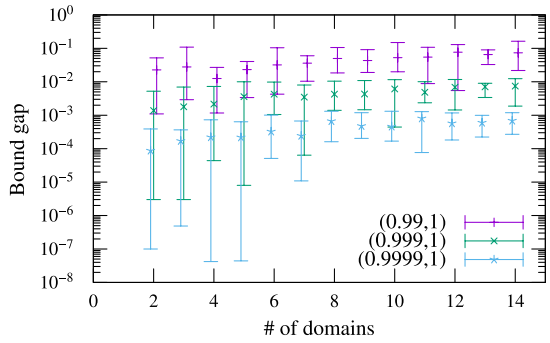


Fig. 12 Bound gaps versus the number of domains.

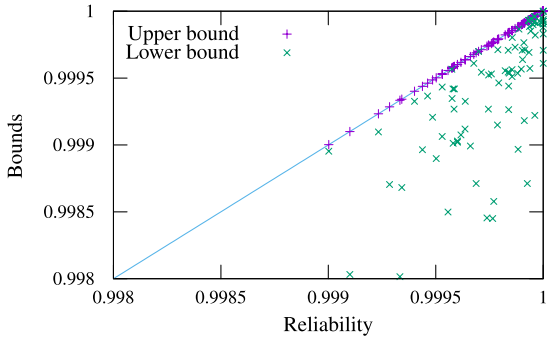


Fig. 13 Lower and upper bounds versus the exact reliability for link availabilities in (0.9999,1).

which could exclude several connected subgraphs. Note that while the upper bounds seem coincident with exact values, they are not; the discussion in Sect. 3.3 explains that upper bounds are tight.

Although our method has errors, they are bounded. This is a key advantage of our method against the existing sampling approach [4]; the bounds allow us to confidently judge whether the network is reliable enough.

6. Related Work

No work has investigated the intra-domain privacy issue in the context of network reliability evaluation. In this respect, our method is the only approach for evaluating the reliability of multi-domain networks. In this section, we summarize related work for reliability evaluation without considering intra-domain privacy and also discuss it for multi-domain networks in general.

Several methods to compute network reliability have been proposed including sum-of-disjoint products [26], factoring theorem [27], decomposition method [28], and binary decision diagrams [14]–[16], [29]. They compute the exact reliability without partitioning the problem. No work has succeeded in computing the reliability of real networks with more than 200 links. Our method utilizes these methods to solve sub-problems defined in our theory, so as to yield lower and upper bounds of the reliability. As shown in Sect. 5.1, our method outperforms the state-of-the-art of exact methods in terms of computation costs.

Sampling approaches like Monte Carlo simulations [4] scale well, but the solution can deviate significantly [15], [17]. Reference [30] proposes F-Monte Carlo; it estimates the probability of rare events accurately, but it depends on the unrealistic assumption that all links would fail with *equal* probability. The most critical issue of this approach is that no guarantee is given as to solution accuracy. Non-guaranteed reliability could cause unexpected disruption of the key social infrastructure. Our method provides error bounds, which guarantee the solution accuracy.

The privacy issue has not been studied in the long history of network reliability. This will be a key issue in the future of network services, since multi-domain services have been recently discussed in the standardization bodies [10], [11]. Minimum-cost networks can be constructed securing intra-domain privacy [8], [9], but the reliability has not been studied. We believe that our work opens a new direction in the research of network reliability.

7. Conclusion

This paper proposed a method to compute lower and upper bounds of reliability for multi-domain networks, without disclosing intra-domain information. The problem is partitioned into subproblems for each domain, which are privately solved by each domain. The partial results, collected using secure computation techniques, are processed to yield the bounds. Experiments indicated that our method scales very well to support 14 domains with 907 links. The bound gaps are less than 0.001 with high availability links of 0.9999.

It is worth noting that our theory in Sect. 3 does not depend on the communication network, so it could be used to reduce the computation complexity for general graphs given a small vertex separator that breaks the graph into small subgraphs.

In future work, we will elaborate our protocol for one or more specific services. With regard to technical aspects, we will consider directed links, each of which has different availabilities. Since directed links can be handled in a single-domain network [31], we will extend it to multi-domain networks. Node failures and dependent failures have been studied for a single-domain network as well [16], [32], [33], so they could also be extended to the multi-domain scenario.

Acknowledgments

We would like to thank Toru Mano for insightful comments on the protocol design.

References

[1] F. Moskowitz, "The analysis of redundancy networks," Trans. AIEE, Part I: Comm. Electron., vol.77, no.5, pp.627–632, 1958.
[2] L. Fratta and U. Montanari, "A Boolean algebra method for computing the terminal reliability in a communication network," IEEE Trans. Circuit Theory, vol.20, no.3, pp.203–211, 1973.

[3] F.T. Boesch, A. Satyanarayana, and C.L. Suffel, "A survey of some network reliability analysis and synthesis results," Networks, vol.54, no.2, pp.99–107, 2009.
[4] I.B. Gertsbakh and Y. Shpungin, Models of Network Reliability: Analysis, Combinatorics, and Monte Carlo, CRC Press, 2009.
[5] P. Agarwal, A. Efrat, S. Ganjugunte, D. Hay, S. Sankararaman, and G. Zussman, "The resilience of WDM networks to probabilistic geographical failures," IEEE/ACM Trans. Netw., vol.21, no.5, pp.1525–1538, 2013.
[6] H. Saito, "Geometric evaluation of survivability of disaster-affected network with probabilistic failure," Proc. IEEE INFOCOM, pp.1608–1616, 2014.
[7] F. Samuel, M. Chowdhury, and R. Boutaba, "PolyViNE: Policy-based virtual network embedding across multiple domains," J. Internet Services and Applications, vol.4, no.1, pp.1–23, 2013.
[8] T. Mano, T. Inoue, K. Mizutani, and O. Akashi, "Virtual network embedding across multiple domains with secure multi-party computation," IEICE Trans. Commun., vol.E98-B, no.3, pp.437–448, March 2015.
[9] T. Mano, T. Inoue, D. Ikarashi, K. Hamada, K. Mizutani, and O. Akashi, "Efficient virtual network optimization across multiple domains without revealing private information," IEEE Trans. Netw. Serv. Manage., vol.13, no.3, pp.477–488, 2016.
[10] "Network functions virtualization – white paper on NFV priorities for 5G," ETSI ISG NFV, 2017.
[11] C.J. Bernardos, L.M. Contreras, I. Vaishnavi, R. Szabo, X. Li, F. Paolucci, A. Sgambelluri, B. Martini, L. Valcarenghi, G. Landi, D. Andrushko, and A. Mourad, "Multi-domain network virtualization," Internet Engineering Task Force, Internet-Draft draft-bernardos-nfvrg-multidomain-05, 2018, work in progress.
[12] L.G. Valiant, "The complexity of enumeration and reliability problems," SIAM J. Comput., vol.8, no.3, pp.410–421, 1979.
[13] M.O. Ball, "Computational complexity of network reliability analysis: An overview," IEEE Trans. Rel., vol.35, no.3, pp.230–239, 1986.
[14] M. Lê, M. Walter, and J. Weidendorfer, "Improving the Kuo-Lu-Yeh algorithm for assessing two-terminal reliability," Proc. European Dependable Computing Conference, pp.13–22, 2014.
[15] T. Inoue, "Reliability analysis for disjoint paths," IEEE Trans. Rel., vol.68, no.3, pp.985–998, 2019.
[16] J. Kawahara, K. Sonoda, T. Inoue, and S. Kasahara, "Efficient construction of binary decision diagrams for network reliability with imperfect vertices," Reliab. Eng. Syst. Safe., vol.188, pp.142–154, 2019.
[17] M. Nishino, T. Inoue, N. Yaasuda, S. Minato, and M. Nagata, "Optimizing network reliability via best-first search over decision diagrams," Proc. IEEE Conference on Computer Communications, ser. INFOCOM, pp.1817–1825, 2018.
[18] D.E. Knuth, The Art of Computer Programming, Volume 4A, Combinatorial Algorithms, Part 1, 1st ed., Addison-Wesley Professional, 2011.
[19] ETSI GR NFV-IFA 022 V3.1.1 Network Functions Virtualisation(NFV) Release 3; Management and Orchestration; Report on Management and Connectivity for Multi-Site Services, ETSI ISG NFV, 2018.
[20] ETSI GR NFV-IFA 028 V3.1.1 Network Functions Virtualisation(NFV) Release 3; Management and Orchestration; Report on architecture options to support multiple administrative domains, ETSI ISG NFV, 2018.
[21] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang, "BGP routing stability of popular destinations," Proc. 2nd ACM SIGCOMM Workshop on Internet Measurment, ser. IMW'02, pp.197–202, 2002.
[22] O. Goldreich, Foundations of Cryptography: Volume 2, Basic Applications, Cambridge University Press, 2004.
[23] C. Gentry, A Fully Homomorphic Encryption Scheme, Ph.D. dissertation, aAI3382729, Stanford, CA, USA, 2009.
[24] S. Knight, H.X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The Internet topology zoo," IEEE J. Sel. Areas Commun., vol.29, no.9, pp.1765–1775, 2011.
[25] T. Inoue, H. Iwashita, J. Kawahara, and S. Minato, "Graphillion: Software library for very large sets of labeled graphs," International Journal on Software Tools for Technology Transfer, vol.18, no.1, pp.57–66, 2016.
[26] J.M. Wilson, "An improved minimizing algorithm for sum of disjoint products (reliability theory)," IEEE Trans. Rel., vol.39, no.1, pp.42–45, 1990.
[27] A. Satyanarayana and M.K. Chang, "Network reliability and the factoring theorem," Networks, vol.13, no.1, pp.107–120, 1983.
[28] J. Carlier and C. Lucet, "A decomposition algorithm for network reliability evaluation," Discrete Appl. Math., vol.65, no.1, pp.141–156, 1996, first International Colloquium on Graphs and Optimization.
[29] K. Sekine, H. Imai, and S. Tani, "Computing the Tutte polynomial of a graph of moderate size," Proc. International Symposium on Algorithms and Computation, pp.224–233, 1995.
[30] E. Canale, F. Robledo, P. Romero, and P. Sartor, "Monte Carlo methods in diameter-constrained reliability," Optical Switching and Networking, vol.14, pp.134–148, 2014.
[31] T. Maehara, H. Suzuki, and M. Ishihata, "Exact computation of influence spread by binary decision diagrams," Proc. International Conference on World Wide Web, pp.947–956, 2017.
[32] T. Yoshida, J. Kawahara, T. Inoue, and S. Kasahara, "On network reliability with link failure dependencies using bdds," IPSJ, Technical Report, 2017-AL-1653, 2017 (in Japanese).
[33] J. Kawahara, T. Inoue, and S. Kasahara, "Network reliability evaluation with arbitrary dependencies on link failures," IEICE, Technical Report, CQ2018-113, 2019 (in Japanese).

Atsushi Taniguchi is a senior research engineer at the NTT Network Innovation laboratories and also a Ph.D. student at the SOKENDAI. He received his B.S. and M.S. degrees in applied physics and chemistry from University of Electro-Communications (UEC) in 2001 and 2003, respectively. He was a research fellow at National Institute of Information and Communications Technology (NICT) from 2006 to 2008, and worked at NTT Communications from 2008 to 2014. His research interests network virtualization and management technologies. He received the best paper award from OptoElectrinics and Communications Conference (OECC) in 2007.

Takeru Inoue is a senior researcher in NTT Laboratories. He was an ERATO researcher at the Japan science and technology agency from 2011 through 2013. His research interests widely cover the design and control of network systems. He received the best paper award from the Asia-Pacific conference on communications in 2005, and also the research awards of the IEICE Information Network Group in 2002, 2005, 2012, and 2015. He received the B.E., M.E., and Ph.D. degrees from Kyoto University, Kyoto, Japan, in 1998, 2000, and 2006, respectively. He is a member of IEEE.

Kohei Mizuno received the B.E. and M.E. degrees from Keio University, Kanagawa, Japan, in 1997 and 1999, respectively. In 1999, he joined NTT (Nippon Telegraph and Telephone Corp.) Network Innovation Laboratories, where he was engaged in research and development of an Active RFID Tags, Home ICT and High reliable radio system. He is now researching a Network Softwarization. He is now Senior Research Engineer, Supervisor and Group Leader in NTT Network Innovation Laboratories. He received the IEICE young researcher's award in 2005. He is a member of IEICE and IEEE.

Takashi Kurimoto graduated from the Tokyo Institute of Technology, Japan, where he received B.E. and M.E. degrees in applied physics 1992 and 1994, respectively. In 1994, He graduated from Keio University where he received the Ph.D. degree in 2012. he worked for NTT Network Service Systems Laboratories and NTT east plant planning department from 1994 to 2014. He has been engaged in researching the switching technology for high-speed computer networks and deployment of the next generation network. He moved to NII in 2015 and is currently involved in the design and implementation of the Science Information Network (SINET). He received the IEICE Switching System Research Award in 1996.

Atsuko Takefusa received her B.S., M.S., and Ph.D. (Sci.) degrees from the Ochanomizu University in 1996, 1998, and 2000, respectively. She worked at the Ochanomizu University, the National Institute of Advanced Industrial Science and Technology (AIST),and joined the National Institute of Informatics as an associate professor in 2016. Her research field is parallel and distributed computing including Grid, Cloud and HPC. She is a member of ACM, IEEE, IPSJ and IEICE.

Shigeo Urushidani is a deputy director general and a professor of the National Institute of Informatics (NII), Japan. He is also the director at Cyber Science Infrastructure Development Department of NII and a professor of SOKENDAI. He received B.E. and M.E. degrees from Kobe University in 1983 and 1985, respectively, and received a Ph.D. from the University of Tokyo in 2002. He worked for NTT from 1985 to 2006, where he was engaged in the research and development of ATM, AIN, IP/MPLS, and optical switching systems. He moved to NII in 2006 and is currently involved in the design and implementation of the Science Information Network (SINET), as well as in the research and development on future network architectures.