Robust Routing in Interdependent Networks

Jianan Zhang and Eytan Modiano
Laboratory for Information and Decision Systems, Massachusetts Institute of Technology

***Abstract*—We consider a model of two interdependent networks, where every node in one network depends on one or more supply nodes in the other network and a node fails if it loses all of its supply nodes. We develop algorithms to compute the failure probability of a path, and obtain the most reliable path between a pair of nodes in a network, under the condition that each supply node fails independently with a given probability. Our work generalizes the classical shared risk group model, by considering multiple risks associated with a node and letting a node fail if all the risks occur. Moreover, we study the diverse routing problem by considering two paths between a pair of nodes. We define two paths to be d-failure resilient if at least one path survives after removing d or fewer supply nodes, which generalizes the concept of disjoint paths in a single network, and risk-disjoint paths in a classical shared risk group model. We compute the probability that both paths fail, and develop algorithms to compute the most reliable pair of paths.**

I. Introduction

Many modern systems are interdependent, such as smart power grids, smart transportation, and other cyber-physical systems [1], [2], [3], [4], [5]. In interdependent networks, one network depends on another to properly function. For example, in smart grids, power generators rely on messages from the control center to adjust to the power demands, while the control center relies on the electric power to operate. Due to the interdependence, failures in one network may cascade to another. It is important to understand the robustness of interdependent networks which are prone to cascading failures.

Most previous studies on interdependent networks have focused on the network connectivity based on random graph models, in the asymptotic regime where the number of nodes approaches infinity [6], [7]. The finite-size arbitrary-topology graph models, which represent real communication and physical networks, have been largely overlooked in the interdependent networks literature. A few exceptions include [4], [5], which model interdependent power grids and communication networks by graphs with topologies specified by the real networks. Similarly, we abstract interdependent networks by graphs with specified topologies, which can be tailored for a wide range of applications.

In this paper, we study robust routing problems in interdependent networks, by characterizing the effects of failures in one network on the other network. For an overview of the problems and challenges, it is helpful to consider a simplified scenario where a demand network depends on a supply network, illustrated by Fig. 1. Every node in the demand network is supported by one or more nodes in the supply network. Thus, nodes in the demand network and nodes in the supply network can be viewed as demand nodes and supply nodes, respectively. Given that a demand node fails if it loses all of its supply nodes, supply node failures may lead to correlated demand node failures, which makes it difficult to route traffic through reliable paths in the demand network. We develop techniques to tackle the failure correlation. This simplified one-way dependence exists in current systems. For example, routers and processors in a communication network depend on the electric power. Moreover, as we will see later, the analysis based on this simplified scenario can be applied to interdependent networks under certain assumptions.

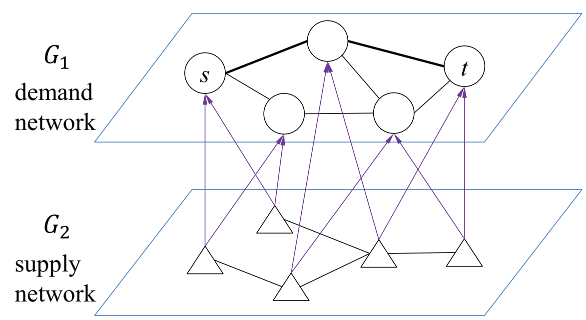

Fig. 1. Every node in the demand network G_1 is supported by two nodes in the supply network G_2.

The robust routing problems have been extensively studied under both independent failure and correlated failure scenarios. If edges or nodes fail independently, the most reliable path between a source-destination pair can be viewed as a shortest path, where the length is a function of the failure probability. In the case of correlated failures, it is difficult to find a path with any performance guarantee in general [8]. If correlation only exists among edges or nodes that fail simultaneously, the network can be viewed using a shared risk group model (Fig. 2) [9], [10]. The shared risk group model captures correlated failures in an overlay network when underlay failures occur, and is commonly used to study the cross-layer reliability, such as logical link failures caused by fiber failures in optical networks [11], [12], [13], [14]. The most reliable path contains the smallest number of risks if all risks are equally likely to occur, and can be obtained by integer programming [11].

Interdependent networks have similarities with the classical shared risk group model, in that two demand nodes share a risk if they have at least one common supply node. However,

This work was supported by DTRA grants HDTRA1-13-1-0021 and HDTRA1-14-1-0058.

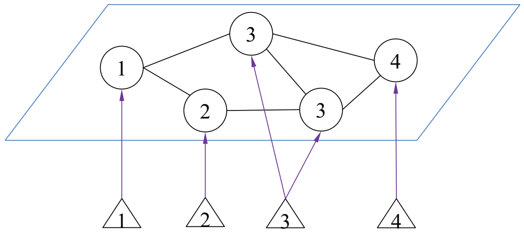

Fig. 2. A shared risk node group model, where nodes labeled by the same number share the same risk.

the key difference is that a demand node does not necessarily fail if a risk occurs (*i.e.*, a supply node fails), since a demand node may have multiple supply nodes, whereas a node fails if its associated risk occurs in the classical shared risk group model.

The reliability of a path in interdependent networks, in contrast to the classical shared risk group model, can no longer be characterized by the number of risks that the path contains. For example, if all the nodes in a path depend on a single supply node and thus the path has a single risk, removing a single supply node would disconnect the path. In contrast, if every node in a path has multiple supply nodes, the path would be more robust and can resist a larger number of supply node failures, although the path has more "risks".

In addition to the most reliable path, a backup path can be used to further improve reliability, through *diverse routing*. Intuitively, a pair of reliable paths should share the minimum number of risks (or be risk-disjoint) in the shared risk group model [11], [13]. However, in interdependent networks, it is easy to construct examples where two paths that share many supply nodes can withstand a larger number of supply node failures than two paths that share a smaller number of supply nodes (*e.g.*, Fig. 3). New metrics, other than the number of risks shared by two paths, need to be identified to characterize their reliability.

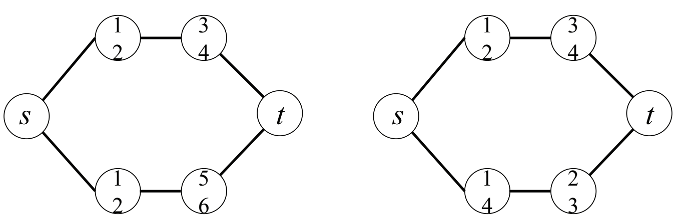

Fig. 3. The two numbers in each node represent its two supply nodes. The two st paths in the left figure share two supply nodes, and can be both disconnected after removing supply nodes $\{1, 2\}$. The two st paths in the right figure share four supply nodes, but they cannot be both disconnected after removing any two supply nodes.

Diverse routing problems have been studied under correlated link failures. The correlation between a pair of logical links is obtained either by measurement [15] or by analysis of the underlay physical topology [16]. Heuristic algorithms have been developed to find multiple reliable paths, and their performance was evaluated by simulation [15], [17], [18]. In contrast, we explicitly bound the gap between the failure probability and the optimization objective, and develop algorithms that have provable performance.

In this paper, we develop an analytically tractable framework to study the following robust routing problems in interdependent networks.

`Single-path routing:` Compute the probability that a specified path fails. Obtain the most reliable path between a source-destination pair.

`Diverse routing:` Compute the probability that two specified paths both fail. Obtain the pair of most reliable paths between a source-destination pair.

By generalizing the concept of disjoint paths to interdependent networks, we characterize the level of disjointness between two paths to study diverse routing. In contrast to the classical shared risk group model where a node fails if its risk occurs, in interdependent networks a node fails if a combination of risks occur. In view of this, our methods extend the shared risk group model. To the best of our knowledge, this paper is the first to study the robust routing problem in interdependent networks.

The rest of the paper is organized as follows. In Section II, we state our model for interdependent networks and failures. In Section III, we prove the complexity, and develop approximation algorithms to compute the path failure probability. In Section IV, we develop algorithms to find the most reliable path between a pair of nodes. In Section V, we study the diverse routing problem in interdependent networks, and find a pair of reliable paths whose failure probability is minimized. Section VI provides numerical results. Finally, Section VII concludes the paper.

II. Model

We consider a demand network G_1 and a supply network G_2, where every demand node in G_1 depends on one or more supply nodes in G_2. We assume that every supply node provides substitutional supply to the demand nodes, and a demand node is functioning if it is directly connected to at least one supply node. To study the impact of node failures in G_2 on G_1, it is equivalent to study the following model.

Consider a graph $G(V, E, \mathcal{S}_V)$, where nodes V and edges E are identical to nodes and edges in G_1, and $\mathcal{S}_V$ are the supply node sets, each of which is a set of nodes in G_2 that provide supply to a node in V. In this model, each node $v_i \in V$ is a demand node, supported by a set of supply nodes $S_i \in \mathcal{S}_V$, and v_i fails if all the nodes in S_i fail. (Note that nodes V may have different number of supply nodes.) Finally, let $s, t \in V$ be a source-destination pair.

Under the condition that supply nodes fail independently with given probabilities, and following the convention that s, t do not fail, we study the robust routing problems in $G(V, E, \mathcal{S}_V)$.

Remark. The analysis for this model can be directly applied to interdependent networks, as long as the interdependence is bidirectional (*i.e.*, if $v \in G_1$ depends on $u \in G_2$, then u depends on v as well) and failures initially occur in one network. It suffices to observe that, given a set of failed nodes $S \subseteq G_2$, a node $v \in G_1$ fails if and only if its supply nodes are all in S. Notice that the failure of v does not further lead to node failures in G_2, because all the nodes that v supports, which are exactly the supply nodes for v due to the bidirectional interdependence, have failed.

III. Computing the reliability of a path

If every node has a single supply node, the path failure probability is given by $1-(1-p)^r$, where each supply node fails independently with probability p and the path is supported by r supply nodes. In contrast, if every node has more than one supply node, computing the path failure probability becomes $\#P$-hard. The proof can be found in the Appendix of the technical report [19].

Theorem 1. *Computing the failure probability of a path is $\#P$-hard, if every node has two or more supply nodes and each supply node fails independently with probability p.*

Although it is $\#P$-hard to compute, the path failure probability can be well approximated. We apply the solution to the *DNF probability* problem and propose an (ϵ, δ)-approximation algorithm based on importance sampling, which approximates the path failure probability to within a multiplicative factor $1 \pm \epsilon$ with probability at least $1-\delta$.

The DNF probability problem computes the probability that a Disjunctive Normal Form (DNF) formula is true, when literals are set to be true independently with given probabilities. A DNF formula is a disjunction of clauses, each of which is a conjunction of literals, and takes the following form: $(x_1^1 \wedge \cdots \wedge x_{n_1}^1) \vee (x_1^2 \wedge \cdots \wedge x_{n_2}^2) \vee \cdots \vee (x_1^m \wedge \cdots \wedge x_{n_m}^m)$. Let $v_1 - \cdots - v_m$ be a path in $G(V, E, \mathcal{S}_V)$. The key observation is that computing the path failure probability can be formulated by a DNF probability problem, in which a clause C_i represents a node v_i in the path and the literals x_j^i in clause C_i represent the supply nodes of v_i. For completeness, we state Algorithm 1 that approximates the path failure probability, by adapting the algorithm that approximates the DNF probability in [20].

The intuition behind this importance sampling algorithm is as follows. Some events, although rare, are important in determining the path failure probability, especially when the path failure probability is small. The algorithm samples in a space consisting of important events, each of which is a set of supply node failures U that lead to the path failure. In this space, the failure of U may appear multiple times, given that multiple choices of v_i in Step 2 may lead to the same U in Step 3. The algorithm then remove the duplicated U via sampling in Step 4.

To prove the correctness of the algorithm, we take the following two steps. First, following a similar analysis to [20], we prove that the path failure probability is given by $E[I] \sum_{1 \le k \le m} \prod_{1 \le j \le n_s(v_k)} p(u_j^k)$, where $E[I]$ is the expectation of I in Step 4 of the algorithm. Second, by repeating the loop a sufficiently large number of times, $E[I]$ can be approximated to within factor $1 \pm \epsilon$ with probability at least $1-\delta$. The details of the proof can be found in the Appendix of the technical report [19].

Algorithm 1 Estimating the path failure probability based on importance sampling.

Initialization:

1) Given a path $\{v_1, v_2, \ldots, v_m\}$, let $\{u_j^i | j = 1, \ldots, n_s(v_i)\}$ denote the set of supply nodes of v_i, where $n_s(v_i)$ is the number of supply nodes of v_i.

Main loop:

2) Among $\{v_1, v_2, \ldots, v_m\}$, randomly choose v_i with probability
$$\prod_{1 \le j \le n_s(v_i)} p(u_j^i) / \sum_{1 \le k \le m} \prod_{1 \le j \le n_s(v_k)} p(u_j^k).$$
If every demand node has an identical number of supply nodes, and the supply node failure probability $p(u_j^i)$ is identical, then node v_i is chosen with probability $1/m$.
3) If v_i is chosen, set all of its supply nodes $\{u_j^i | j = 1, \ldots, n_s(v_i)\}$ to be failed. The other supply nodes are randomly set to be failed with their respective failure probabilities. Let U denote the set of failed supply nodes.
4) Test whether v_i is the first failed node among $\{v_1, v_2, \ldots, v_m\}$, given that U fail (and no other supply nodes fail). If true, set $I = 1$; otherwise, set $I = 0$. Repeat the loop for $a = 3m \ln(2/\delta)/\epsilon^2$ iterations.

Result:

5) Count the number of $I = 1$ and denote the number by b. An (ϵ, δ)-approximation of the path failure probability is given by $b/a \sum_{1 \le k \le m} \prod_{1 \le j \le n_s(v_k)} p(u_j^k)$.

The advantage of this algorithm over a naïve Monte-Carlo algorithm (*e.g.*, by repeatedly simulating the supply node failure events and counting the fraction of trials in which the path fails) is that the number of iterations in the naïve Monte-Carlo algorithm is large when the path failure probability is small[1]. In contrast, by sampling in a more important space, the number of iterations is reduced. Note that the only quantity that needs to be estimated in Algorithm 1 by simulation is $E[I]$, and that $\Pr(I = 1) \ge 1/m$. We conclude this section by the following theorem, whose proof is in the Appendix of the technical report [19].

Theorem 2. *The path failure probability can be estimated to within a multiplicative factor $1 \pm \epsilon$ with probability $1-\delta$, in time $O(m^2 n_s \ln(1/\delta)/\epsilon^2)$, where m is the path length and n_s is the maximum number of supply nodes for a demand node.*

Although the failure probability of a specific path can be well approximated by the importance sampling algorithm, the algorithm hardly gives an intuition for path properties that characterize a reliable path. In the remainder of this section, we

[1] If F occurs in b out of a trials, $\Pr(F) \in (1 \pm \epsilon) b/a$ with probability $1-\delta$, under the condition that $b = \Omega(\ln(1/\delta)/\epsilon^2)$. The total number of trials $a = \Omega(\ln(1/\delta)/\epsilon^2)/\Pr(F)$ is large when $\Pr(F)$ is small.

develop indicators and bounds on the path failure probability, which can be used for finding the most reliable path.

A. Small and identical failure probability

Consider a path $v_1 - \cdots - v_m$ in $G(V, E, \mathcal{S}_V)$. Let F_i denote the event that all the supply nodes of v_i fail. Let F denote the event that the path fails. Clearly, the path fails if at least one node v_i loses all of its supply nodes ($F = \cup_{1 \leq i \leq m} F_i$).

By the inclusion-exclusion principle, we have

$$\Pr(F) = \sum_{1 \leq i \leq m} \Pr(F_i) - \sum_{1 \leq i_1 < i_2 \leq m} \Pr(F_{i_1} \cap F_{i_2}) + \cdots + (-1)^{m-1} \Pr(F_1 \cap F_2 \cdots \cap F_m). \quad (1)$$

Directly computing the path failure probability is difficult, given that there are $\binom{m}{j}$ summations in the j-th term of the inclusion-exclusion formula. We first reduce the number of events in the inclusion-exclusion formula, and then further simplify the computation under the condition that the supply node failure probability is small and identical.

To reduce the number of events, some *redundant* events can be ignored. For example, if F_i occurs only if F_j occurs, then the event F_i is redundant in determining F with the knowledge of F_j. To see this, note that 1) if F_j occurs, then the path fails regardless of F_i; 2) if F_j does not occur, then F_i does not occur as well. If the supply nodes of v_j form a subset of the supply nodes of v_i, then F_i is redundant. With an abuse of language, we call a node v_i redundant if F_i (*i.e.*, the state of v_i) is redundant. With this simplification, we derive the following result.

Let $n_s(v_i)$ denote the number of distinct supply nodes of v_i. Let $n_s^{\min} = \min_{1 \leq i \leq m} n_s(v_i)$. After removing the redundant nodes sequentially, let $\bar{m}$ be the number of remaining nodes that each have $n_s^{\min}$ supply nodes. The path failure probability can be estimated by the following theorem.

Theorem 3. *If every supply node fails independently with probability $p \leq \epsilon/m$, then the path failure probability satisfies $(1-\epsilon)\bar{m}p^{n_s^{\min}} \leq \Pr(F) \leq (1+\epsilon)\bar{m}p^{n_s^{\min}}$.*

Proof. We first reduce the number of failure events that appear in the inclusion-exclusion formula by removing the redundant nodes. Note that determining whether a node is redundant and removing the redundant node are done sequentially. Thus, among the set of nodes that have the same supply nodes, one node remains. Let D denote the nodes in the path excluding the redundant nodes.

First, we consider the first term in Eq. (1) that provides an upper bound on the path failure probability, known as the union bound. Let $D_1 \subset D$ denote the set of nodes that each have $n_s^{\min}$ supply nodes, and let $\bar{m} = |D_1|$. The remaining nodes $D_2 = D \setminus D_1$ each have $n_s^{\min} + 1$ or more supply nodes. Thus, the first term of Eq. (1) is at most

$$\begin{aligned} \Pr(F) \leq & \bar{m}p^{n_s^{\min}} + (m - \bar{m})pp^{n_s^{\min}} \\ \leq & \bar{m}p^{n_s^{\min}} + \epsilon p^{n_s^{\min}}, \end{aligned}$$

for $p \leq \epsilon/m$.

Next, we consider the first two terms that provide a lower bound on the path failure probability (*cf*. Bonferroni inequalities). For any pair of nodes $v_j, v_k \in D$, the union of their supply node sets contains at least $\max(n_s(v_j), n_s(v_k)) + 1$ nodes, because neither supply node set includes the other as a subset. At least $n_s^{\min} + 1$ supply nodes have to be removed in order for a pair of nodes in D_1 to fail. At least $n_s^{\min} + 2$ supply nodes need to be removed in order for a pair of nodes to fail if at least one node belongs to D_2. The absolute value of the second term is at most $\binom{\bar{m}}{2} p^{n_s^{\min}+1} + [\binom{m}{2} - \binom{\bar{m}}{2}] p^{n_s^{\min}+2}$. A lower bound on $\Pr(F)$ is

$$\begin{aligned} \Pr(F) \geq & \bar{m}p^{n_s^{\min}} - \left(\frac{\bar{m}^2}{2} pp^{n_s^{\min}} + \frac{m^2}{2} p^2 p^{n_s^{\min}} \right) \\ \geq & \bar{m}p^{n_s^{\min}} - \epsilon \bar{m} p^{n_s^{\min}}, \end{aligned}$$

for $p \leq \epsilon/m$. □

Thus, we have obtained the following two *reliability indicators* for a path. These combinatorial properties are useful in finding a reliable path, which will be studied in the next section.

- $n_s^{\min}$: the minimum number of distinct supply nodes for a node in the path.
- $\bar{m}$: the number of combinations of $n_s^{\min}$ supply node failures that lead to the failure of at least one node in the path.

B. Arbitrary failure probability

In contrast with the case where supply node failure probability is small and identical, it is difficult to characterize the reliability of a path by its combinatorial properties, with limited knowledge of node failure probabilities. Therefore, we obtain bounds on path failure probability that will be useful in finding a reliable path.

First, we develop an upper bound on the path failure probability. Let $p(v_i)$ be the failure probability of node v_i, under the condition that each of its supply nodes u_j^i fails independently with probability $p(u_j^i)$. The path failure probability, under the condition that the failures of V are positively correlated, is no larger than the path failure probability by assuming that the failures of V are independent. The proof can be found in the technical report [19].

Lemma 1. *The failure probability of a path P where a supply node u_j^i fails independently with probability $p(u_j^i)$ is upper bounded by $1 - \prod_{v_i \in P}(1 - p(v_i))$.*

Then, we develop a lower bound on the path failure probability. The intuition is as follows. After replacing a supply node that supports multiple demand nodes by multiple independent supply nodes with sufficiently small failure probability, the path failure probability does not increase. In the original graph $G(V, E, \mathcal{S}_V)$, consider a node $v_i \in V$. Let U^i denote the set of supply nodes of v_i, let $u_j^i \in U^i$ denote one supply node, let $p(u_j^i)$ denote the failure probability of u_j^i, and let $n_d(u_j^i)$ denote the number of nodes that u_j^i supports. Let $\tilde{p}(v_i) = \prod_{u_j^i \in U^i} \tilde{p}(u_j^i)$ denote the failure probability of v_i

if u_j^i fails independently with probability $\tilde{p}(u_j^i) = 1 - (1 - p(u_j^i))^{1/n_d(u_j^i)}$. A lower bound on the path failure probability is as follows, whose proof follows a similar technique in [21] and is in the technical report.

Lemma 2. *The failure probability of a path P where a supply node u_j^i fails independently with probability $p(u_j^i)$ is lower bounded by $1 - \prod_{v_i \in P}(1 - \tilde{p}(v_i))$.*

Let n_d denote the maximum number of demand nodes that a supply node supports, and let n_s denote the maximum number of supply nodes for a demand node. The following lemma bounds the ratio between the upper and lower bounds. Its proof can be found in the Appendix of the technical report [19].

Lemma 3. *For any path, the ratio of the upper bound on its failure probability obtained in Lemma 1 to the lower bound obtained in Lemma 2 is at most $(n_d)^{n_s}$.*

IV. Finding the most reliable path

In this section, we aim to compute the most reliable path between a source-destination pair $s, t \in V$ in $G(V, E, \mathcal{S}_V)$. We first prove that it is NP-hard to approximately compute the most reliable path. We then develop an algorithm to compute the most reliable path when the supply nodes fail independently with an identically small probability, and finally develop an approximation algorithm under arbitrary failure probabilities.

Hardness of approximation: Although the failure probability of any given path can be approximated to within factor $1 \pm \epsilon$ for any $\epsilon > 0$, it is NP-hard to obtain an st path whose failure probability is less than $1 + \epsilon$ times the optimal for a small ϵ. The proof can be found in the Appendix of the technical report [19].

Theorem 4. *Computing an st path whose failure probability is less than $1 + \epsilon$ times the failure probability of the most reliable st path is NP-hard for $\epsilon < 1/m$, where m is the maximum path length.*

A. Small and identical failure probability

If every supply node fails independently with an identically small probability, there are two reliability indicators: $n_s^{\min}$ and $\bar{m}$. Recall that $n_s^{\min}$ is the minimum number of supply nodes for a node in the path, and that $\bar{m}$ is the number of combinations of $n_s^{\min}$ supply node failures that disconnect the path. With the two indicators, the path failure probability can be approximated to within a multiplicative factor $1 \pm \epsilon$ by $\bar{m}p^{n_s^{\min}}$, under the condition that $p \le \epsilon/m$. Moreover, the indicator $n_s^{\min}$ is more important (and has a higher priority to be optimized) than $\bar{m}$. We next develop algorithms to optimize the two indicators.

Given a graph $G(V, E, \mathcal{S}_V)$ and a pair of nodes (s, t), the problem of computing an st path with the maximum $n_s^{\min}$ can be formulated as the *maximum capacity path* problem, where the capacity of a node equals the number of its distinct supply nodes and the capacity of a path is the minimum node capacity along the path. The maximum capacity path can be obtained by a modified Dijkstra's algorithm, and can be obtained in linear time [22].

However, it is NP-hard to minimize $\bar{m}$, even in the special case where every demand node has a single supply node. The result follows from the NP-hardness of computing a path with the minimum colors in a colored graph [12].

We develop an integer program to compute the path P with the minimum $\bar{m}$, under the condition that $n_s^{\min}(P) = \min_{v_i \in P} n_s(v_i)$ is maximized. The following pre-processing reduces the size of the integer program. First, compute $k = \max_{P \in \mathcal{P}} n_s^{\min}(P)$, where $\mathcal{P}$ is the set of all the st paths, using the linear-time maximum capacity path algorithm. Then, remove all the nodes that have fewer than k distinct supply nodes and their attached edges, and denote the remaining graph by $G'(V', E', \mathcal{S}_{V'})$. The removed nodes and edges will not be used by the optimal path. Let $V'' \subseteq V'$ denote the nodes among which each has exactly k distinct supply nodes. We aim to find a path V_P where the number of distinct supply node sets for $V_P \cap V''$ is minimized.

Let S_i denote the set of supply nodes of $i \in V''$. Let $\mathcal{S}_{V''}$ denote the union of these sets. Let x_{ij} denote the flow variable which takes a positive value if and only if edge (i, j) belongs to the selected path. An st path is identified by constraint (3). A node i is on the selected path if at least one of x_{ij} and x_{ji} is positive. Let $h(S_i)$ denote whether removing supply nodes S_i disconnects the selected path. If a node i is on the selected path and has k supply nodes, then $h(S_i)$ must be one, guaranteed by constraint (4). All the other nodes either do not belong to the selected path or have more than k supply nodes, and their supply node failures are not considered. The objective minimizes $\bar{m}$, which is the number of combinations of k supply node failures that disconnect the path.

$$\min \sum_{S_i \in \mathcal{S}_{V''}} h(S_i) \tag{2}$$

$$\text{s.t.} \sum_{\{j|(i,j)\in E'\}} x_{ij} - \sum_{\{j|(j,i)\in E'\}} x_{ji} = \begin{cases} 1, & \text{if } i = s, \\ -1, & \text{if } i = t, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

$$\sum_{\{j|(i,j)\in E'\}} x_{ij} + \sum_{\{j|(j,i)\in E'\}} x_{ji} \le 2h(S_i), \ \forall i \in V'' \setminus s, t, \tag{4}$$

$$x_{ij} \ge 0, \quad \forall (i, j) \in E',$$

$$h(S_i) = \{0, 1\}, \quad \forall S_i \in \mathcal{S}_{V''}.$$

B. Arbitrary failure probability

If nodes $\tilde{V}$ in a graph $\tilde{G}(\tilde{V}, \tilde{E})$ fail independently, the probability that a path survives is the product of the survival probabilities of nodes along the path. The most reliable path can be obtained by the classical shortest path algorithm, by replacing the length of traversing a node $\tilde{v}_i$ by $-\ln(1 - p(\tilde{v}_i))$, where $p(\tilde{v}_i)$ is the failure probability of $\tilde{v}_i$. It is easy to see that the length of a path P is $\sum_{\tilde{v}_i \in P} - \ln(1 - p(\tilde{v}_i)) = -\ln \prod_{\tilde{v}_i \in P}(1 - p(\tilde{v}_i))$. The shortest path has the smallest failure probability $1 - \prod_{\tilde{v}_i \in P}(1 - p(\tilde{v}_i))$.

Compared with the above simple model, the difficulty in obtaining the most reliable st path in interdependent networks is the failure correlations of nodes $V \subseteq G(V, E, \mathcal{S}_V)$. The failure probability of a path can no longer be characterized by $1 - \prod_{v_i \in P}(1 - p(v_i))$. Moreover, let $s - \cdots - v_i - \cdots - t$ be the most reliable st path. The sub-path $s - \cdots - v_i$ may not be the most reliable path between s and v_i. Thus, the label-correction approach in dynamic programming (*e.g.*, Dijkstra's algorithm) cannot be used, even though the failure probability of a given path can be approximated.

Given the bounds obtained in the previous section, we propose Algorithm 2 to compute a path whose failure probability is within $(n_d)^{n_s}$ times the optimal failure probability. Recall that the bounds on path survival probability are the product of (original or new) node survival probabilities, which exactly match the path survival probability in the case of independent node failures.

Algorithm 2 An approximation algorithm to compute a reliable st path in $G(V, E, \mathcal{S}_V)$.

1) For each $v_i \in V$, compute $\tilde{p}(v_i)$ as follows. Let u_j^i be a supply node of v_i with failure probability $p(u_j^i)$. If u_j^i supports $n_d(u_j^i)$ nodes, let $\tilde{p}(u_j^i) = 1 - (1 - p(u_j^i))^{1/n_d(u_j^i)}$. Let $\tilde{p}(v_i)$ be the failure probability of v_i if u_j^i fails independently with probability $\tilde{p}(u_j^i)$.
2) Compute the most reliable st path assuming that v_i fails independently with probability $\tilde{p}(v_i)$. The most reliable path can be obtained by a standard shortest path algorithm (*e.g.*, Dijkstra's algorithm), by letting $-\ln(1 - \tilde{p}(v_i))$ be the length of traversing node v_i.

Theorem 5. *The failure probability of the path obtained by Algorithm 2 is at most $(n_d)^{n_s}$ times the failure probability of the most reliable st path under arbitrary supply node failure probabilities.*

Proof. Let the path obtained by Algorithm 2 be P' and let the path with the minimum failure probability be P^*. Let $p(P')$ and $p(P^*)$ denote their failure probabilities. Moreover, let $\tilde{p}(P')$ and $\tilde{p}(P^*)$ denote their failure probabilities by assuming that each node v_i fails independently with probability $\tilde{p}(v_i)$. We have $p(P') \leq n_d^{n_s}\tilde{p}(P') \leq n_d^{n_s}\tilde{p}(P^*) \leq n_d^{n_s} p(P^*)$, where the first inequality follows from Lemma 3 and the last inequality follows from Lemma 2. $\square$

Remark. If $n_s = 1$ and every supply node fails independently with an identically small probability, our result reduces to the following result in the classical shared risk group model: The number of risks associated with the shortest path is at most n_d times the number of risks associated with the minimum-risk path [13].

V. Reliability of a pair of paths

To study diverse routing in interdependent networks, we consider the simplest case of two st paths in this section. Given that computing the failure probability of a single path is $\#P$ hard if every node has more than one supply node, it is also $\#P$ hard to compute the failure probability of two paths[2]. To see this, note that if two paths have the same number of nodes and each node in the first path has identical supply nodes as its corresponding node in the second path, then the probability that both paths fail equals the probability that a single path fails. Fortunately, we are still able to obtain $1 \pm \epsilon$-approximation of the failure probability in polynomial time.

A. Small and identical failure probability

A central concept in diverse routing is the *disjoint paths* or *risk disjoint paths* [11], [12], [10]. In the classical shared risk group model, if every risk occurs independently with an identically small probability $p = o(1/m^2)$, the probability that two paths fail is $\Theta(f(m)p^2)$ if they are risk disjoint and $\Theta(f(m)p)$ if they share one or more risks, where m is the maximum path length and $f(m)$ is a function of m. Thus, risk-disjointness characterizes the order of the reliability of two paths. In interdependent networks where every demand node has multiple supply nodes, if nodes in P^1 do not share any supply nodes with nodes in P^2, then P^1 and P^2 are risk disjoint. However, risk-disjointness does not suffice to characterize the reliability of two paths, for the following two reasons. First, the failure probability of a demand node depends on the number of supply nodes for it, which is not related to risk-disjointness. Second, if P^1 and P^2 share some supply nodes, the failure probability depends further on the maximum number of supply node failures that the two paths can withstand. To study the reliability of two paths in interdependent networks, we define *d-failure resilient paths* as follows.

Definition 1. Two paths are d-failure resilient if removing any d supply nodes would not disconnect both paths.

Remark. In the classical graph model $\tilde{G}(\tilde{V}, \tilde{E})$, two disjoint paths are 1-failure resilient while two overlapping paths are 0-failure resilient. In the classical shared risk group model, two risk disjoint paths are 1-failure resilient while two paths that share risks are 0-failure resilient. Two paths can never be more than one failure resilient. Thus, the disjointness or risk-disjointness suffices to characterize (the order of) the reliability of two paths in these models.

1) Evaluation of failure probability: Consider two paths $P^1 = s - v_1^1 - v_2^1 - \cdots - v_{m_1}^1 - t$, $P^2 = s - v_1^2 - v_2^2 - \cdots - v_{m_2}^2 - t$ between a pair of nodes (s, t). We study the event that at least one node in P^1 and at least one node in P^2 both fail. Let F_i^k denote the event that all the supply nodes of v_i^k fail, and let F^k denote the event that the k-th path fails, $k \in \{1, 2\}$. Then $F^1 \cap F^2 = \cup_{1 \leq i \leq m_1, 1 \leq j \leq m_2}(F_i^1 \cap F_j^2)$. For simplicity of presentation, let $F^{\text{both}} = F^1 \cap F^2$ and $F_{ij} = F_i^1 \cap F_j^2$. Let S_{ij} denote the union of supply nodes of v_i^1 and v_j^2.

[2]Meanwhile, it is still simple to compute the failure probability of two paths if every node has a single supply node, by first computing the probability that the first path fail, and then computing the probability that the second path fail while the first path does not fail (*i.e.*, none of the supply nodes of the first path fail), both in polynomial time, and summing the two probabilities.

To decide whether two paths are d-failure resilient, we consider the number of supply node failures that lead to the event F_{ij}, and denote the number by d_{ij}. Then $d = \min_{1\leq i\leq m_1, 1\leq j\leq m_2} d_{ij} - 1$. Moreover, let $\bar{m}$ be the number of pairs of nodes, one from each path, such that each pair of nodes in total have $d+1$ distinct supply nodes and any two pairs do not have the same set of $d+1$ supply nodes. (*I.e.*, $\bar{m}$ combinations of $d+1$ supply node failures each disconnect both paths.) The next theorem formalizes the connection between the reliability of two paths and d. The proof follows the same manner as that for Theorem 3 and can be found in the technical report [19].

Theorem 6. *If every supply node fails independently with probability $p \leq \epsilon/(m_1 m_2)$, then the probability that two d-failure resilient paths with lengths m_1, m_2 both fail satisfies $(1-\epsilon)\bar{m}p^{d+1} \leq \Pr(F^{both}) \leq (1+\epsilon)\bar{m}p^{d+1}$.*

2) Finding the most reliable pair of paths: From Theorem 6, we know that the probability that two d-failure resilient paths both fail is smaller for larger values of d. Moreover, for a fixed d, the failure probability is proportional to $\bar{m}$, the number of combinations of $d+1$ supply node failures that disconnect both paths. We have obtained two reliability indicators for two paths: d and $\bar{m}$.

Unfortunately, computing the pair of st paths that have the maximum d and the minimum $\bar{m}$ are both NP-hard, even in the special case where every demand node has a single supply node. This special case reduces to the classical shared risk group model. In this special case, $d = 1$ if there exist two risk-disjoint paths, and $d = 0$ otherwise. The NP-hardness of determining the existence of two risk-disjoint paths between an st pair has been proved in [11]. Moreover, in this special case, for two paths that share common supply nodes, $\bar{m}$ is the number of overlapping risks between the two paths (*i.e.*, removing any of the $\bar{m}$ supply nodes disconnects both paths). The NP-hardness of the least coupled paths problem, which computes a pair of paths that share the minimum number of risks in the classical shared risk group model, has also been proved in [11].

We develop an integer program to compute a pair of st paths with the maximum d in $G(V, E, \mathcal{S}_V)$. Let variable x^k_{ij} denote whether edge (i,j) is part of the k-th path, and let variable b^k_i denote whether node i is part of the k-th path, $k \in \{1,2\}$. Same as before, let S_i denote the supply nodes of node i. Constraints (6) guarantee that two paths are node-disjoint. Notice that these constraints can be dropped if there is no restriction on the physical disjointness of two paths. Constraints (7) guarantee that at least $d+1$ supply nodes need to be removed in order for one node in each path to fail (*i.e.*, $b^1_i = b^2_j = 1$, $i,j \in V$), where M is a sufficiently large number, *e.g.*, twice the maximum number of supply nodes for a demand node.

$$\max \quad d \tag{5}$$

$$\text{s.t.} \quad \sum_{\{j|(i,j)\in E\}} x^k_{ij} - \sum_{\{j|(j,i)\in E\}} x^k_{ji} = \begin{cases} 1, \text{ if } i = s, \\ -1, \text{ if } i = t, \quad k \in \{1,2\}, \\ 0, \text{ otherwise.} \end{cases}$$

$$\sum_{\{j|(i,j)\in E\}} x^k_{ij} + \sum_{\{j|(j,i)\in E\}} x^k_{ji} \leq 2b^k_i, \quad \forall i \in V, k \in \{1,2\}$$

$$b^1_i + b^2_i \leq 1, \ \forall i \in V \setminus s,t, \tag{6}$$

$$d + 1 \leq |S_i \cup S_j| + M(2 - b^1_i - b^2_j), \forall i,j \in V \setminus s,t, \tag{7}$$

$$x^k_{ij} \in \{0,1\}, \quad \forall (i,j) \in E, k \in \{1,2\},$$

$$b^k_i \in \{0,1\}, \quad \forall i \in V, k \in \{1,2\}.$$

A slightly modified integer program suffices to minimize $\bar{m}$ under the condition that d is maximized. Let $h(S_i \cup S_j)$ denote whether removing the union of supply nodes for i and j disconnects both paths. Constraints (9) guarantee that if i and j belong to two different paths, *i.e.*, $b^1_i = b^2_j = 1$, then $h(S_i \cup S_j) = 1$. Otherwise, $h(S_i \cup S_j) = 0$ in the optimal solution. Let a positive value $w(|S_i \cup S_j|)$ denote its *weight*, which is a decreasing function of the cardinality $|S_i \cup S_j|$. We aim to minimize the total weights of supply node failures that disconnect two paths. In order to guarantee that d is maximized, $w(l)/w(l+1)$ should be sufficiently large for any integer l, *e.g.*, $|V|^2/2$. Since there are at most $|V|(|V|-1)/2$ pairs of nodes, larger d is always preferable and has a higher priority to be optimized over $\bar{m}$.

$$\min \quad \sum_{S_i, S_j \in \mathcal{S}_V} w(|S_i \cup S_j|) h(S_i \cup S_j) \tag{8}$$

$$\text{s.t.} \quad \sum_{\{j|(i,j)\in E\}} x^k_{ij} - \sum_{\{j|(j,i)\in E\}} x^k_{ji} = \begin{cases} 1, \text{ if } i = s, \\ -1, \text{ if } i = t, \quad k \in \{1,2\}, \\ 0, \text{ otherwise.} \end{cases}$$

$$\sum_{\{j|(i,j)\in E\}} x^k_{ij} + \sum_{\{j|(j,i)\in E\}} x^k_{ji} \leq 2b^k_i, \quad \forall i \in V, k \in \{1,2\}$$

$$b^1_i + b^2_i \leq 1, \ \forall i \in V \setminus s,t,$$

$$h(S_i \cup S_j) \geq b^1_i + b^2_j - 1, \quad \forall i,j \in V \setminus s,t, \tag{9}$$

$$x^k_{ij} \in \{0,1\}, \quad \forall (i,j) \in E, k \in \{1,2\},$$

$$h(S_i \cup S_j) \in \{0,1\}, \quad \forall i,j \in V.$$

B. Arbitrary failure probability

1) Evaluation of failure probability: We use a similar importance sampling approach to Algorithm 1 and formulate the problem of computing the failure probability of two paths as a DNF probability problem. A clause C_{ij} represents a pair of nodes v^1_i and v^2_j. Literals in C_{ij} represent the union of supply nodes of v^1_i and v^2_j. A literal is true if and only if the supply node that it represents fails, and the probability that the literal is true is the same as the supply node failure probability. The disjunction of clauses is true if and only if at least one clause is true, in which case both paths fail because at least one node from each path fails. The rest of the computation follows the same manner as Algorithm 1, by replacing a node in Algorithm 1 by a pair of nodes. An (ϵ, δ)-approximation of the failure probability $\Pr(F^{\text{both}})$ can be obtained in $O(m_1^2 m_2^2 n_s \ln(1/\delta)/\epsilon^2)$ time.

2) Finding the most reliable pair of paths: It is more difficult to find two paths that have the smallest failure probability. Recall Theorem 6. The failure probability of two paths is $\Theta(f(m_1, m_2)p^{d+1})$ if they are d-failure resilient when the supply node failure probability p is small, where $f(m_1, m_2)$ is a function of two path lengths. As a corollary of the fact that it is NP-hard to compute two paths that have the maximum level of resilience d, it is also NP-hard to compute two paths whose failure probability is within a factor α from the optimal, where α is any function of the network size. Thus, we develop the following heuristic. After computing the failure probability $\tilde{p}(v_i)$ of a node v_i in Step 1 of Algorithm 2, let $-\ln(1-\tilde{p}(v_i))$ be the length of traversing node v_i, and compute two node disjoint paths with the minimum total lengths. The two paths can be efficiently obtained using a slightly modified shortest augmenting path algorithm [23]. The computation is outlined in Algorithm 3. The reason for the graph transformation in Step 1 is to simplify the computation of a residual graph, to which the shortest augmenting path algorithm can be applied.

Algorithm 3 A heuristic to compute a pair of reliable st path in $G(V, E, \mathcal{S}_V)$.

1) Transform $G(V, E, \mathcal{S}_V)$ with node failure probabilities to a directed graph G' with edge failure probabilities using the standard approach. (Split every node v into v_{in} and v_{out}. Add a directed edge from v_{in} to v_{out}, which has length $-\ln(1-\tilde{p}(v_i))$. Add a directed edge from $v_{1\text{out}}$ to $v_{2\text{in}}$ and a directed edge from $v_{2\text{out}}$ to $v_{1\text{in}}$, both with zero length, if an edge exists between v_1 and v_2 in $G(V, E, \mathcal{S}_V)$.)
2) Compute the shortest path P_1' from s_{out} to t_{in} in G'.
3) Compute the residual graph as follows. Remove all the edges in P_1'. Add a backward edge from v_2' to v_1' with a negated length if an edge from v_1' to v_2' is part of P_1'.
4) Compute the shortest path P_2' from s_{out} to t_{in} in the residual graph.
5) Combine P_1' and P_2' by cycle cancellation. The two paths become node-disjoint and can be mapped to two paths in $G(V, E, \mathcal{S}_V)$.

VI. Numerical results

We study the robust routing problems in the XO backbone communication network with 60 nodes and 75 edges [24], by assuming that the XO nodes are supported by 36 randomly generated supply nodes within the continental US. The XO network topology is depicted in Fig. 4, and the supply nodes are marked as triangles. The x-axis represents the longitude and the y-axis represents the latitude. We do not claim that the XO network needs supply from these randomly generated points, and we use this example only to provide a visualization of the robust routing problems using available data.

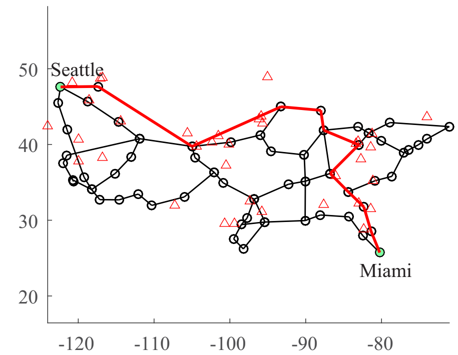

Fig. 4. Topology of the XO network and randomly generated triangle supply nodes. The most reliable Seattle-Miami path is colored red under the condition that supply node failure probability is small and identical and every XO node depends on two nearest supply nodes.

First, we assume that every XO node depends on two nearest supply nodes and every supply node fails independently with probability 10^{-2}. Since the supply node failure probability is small and identical, we are able to obtain the most reliable path and pair of paths by optimizing the reliability indicators using integer programs.

To identify the most reliable path, since $n_s^{\min}(P) = 2$ for any path P, we only need to compute a path with the minimum $\bar{m}$ using the integer program in Section IV. The most reliable path is colored red in Fig. 4, for which $\bar{m} = 8$. To evaluate the path failure probability, by a stronger analog of Theorem 3 (Corollary 1 in the technical report), setting $\epsilon = 4 \times 10^{-2}$, $\Pr(F) \in [7.68 \times 10^{-4}, 8 \times 10^{-4}]$. To compare, using Algorithm 1, we obtain 7.9686×10^{-4} as a (1 ± 0.01)-approximation of the path failure probability with probability 0.99. These results suggest that the two reliability indicators $(n_s^{\min}, \bar{m})$ well characterize the path failure probability when the supply node failure probability is small and identical.

We compute the most reliable pair of paths connecting Seattle-Miami using the integer programs in Section V. The two paths are plotted in Fig. 5, and they are 1-failure resilient ($d = 1$). The failure probability of both paths is approximately 1.0388×10^{-4}. In contrast, the most reliable pair of paths connecting Seattle-Denver are 3-failure resilient and their failure probability is approximately 2.9800×10^{-8}. Thus, the level of resilience well indicates the reliability of two paths.

Next, we assume that an XO node depends on N_s randomly chosen supply nodes, where N_s is uniformly chosen among 1, 2, and 3. Let the failure probability of each supply node be uniformly and independently chosen from $[0.005, 0.015]$. We use Algorithm 2 to obtain a reliable path connecting Seattle-Miami. Averaged over 10 trials, the path failure probability is approximately 2.1032×10^{-2}, while the lower bound on the failure probability of the most reliable path is 5.2365×10^{-3}. The obtained path has failure probability around four times the lower bound. Moreover, by using the heuristic to find a pair of paths, the paths have average failure probability 3.9732×10^{-3}, which improves the reliability of a single path.

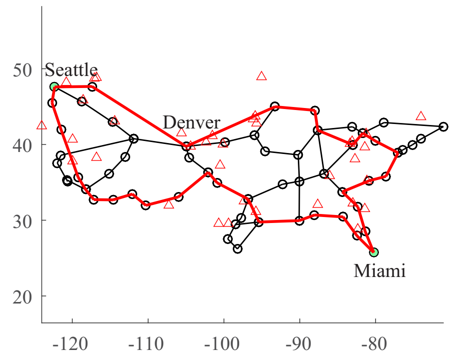

Fig. 5. The most reliable pair of paths between Seattle-Miami are colored red, under the condition that supply node failure probability is small and identical and every XO node depends on two nearest supply nodes.

We compare the performance of the heuristic (Algorithm 3) with the optimal pair of paths. Since it is difficult to obtain the optimal pair of paths under arbitrary failure probabilities, we use the integer program (8), under the condition that supply nodes fail independently with probability 10^{-2}. If every XO node depends on two nearest supply nodes, the failure probabilities of two optimal paths and two paths obtained by the heuristic are approximately 1.0388×10^{-4} and 1.0773×10^{-4}, respectively. If every XO node depends on three nearest supply nodes, the failure probability of two optimal paths and two paths obtained by the heuristic are approximately 1.0200×10^{-6} and 1.0508×10^{-6}, respectively. These experiments validate the performance of our heuristic algorithm.

Finally, we report the running times of the algorithms, executed in a workstation that has an Intel Xeon Processor (E5-2687W v3) and 64GB RAM. The integer programs that find the most reliable path and pair of paths (under small and identical supply node failure probability) can both be solved within 1 second. The approximation algorithm to find a reliable path and the heuristic to find a pair of paths (under arbitrary failure probabilities) can both be solved within 0.1 second. The evaluation of the failure probability of one path or a pair of paths by Algorithm 1 takes several minutes, by setting $\epsilon = \delta = 0.01$. Thus, the algorithms (integer programs and Algorithms 2 and 3) can be used to find reliable routes in realistic size networks.

VII. Conclusion

We studied the robust routing problem in interdependent networks. We developed approximation algorithms to compute the path failure probability, and identified reliability indicators for a path, based on which we develop algorithms to find the most reliable route in interdependent networks. We also studied diverse routing in interdependent networks, and developed approximation algorithms to compute the probability that two paths both fail and to find two reliable paths. Our work extends the shared risk group models, and provides a new framework to study robust routing problems in interdependent networks.

References

[1] O. Yagan, D. Qian, J. Zhang, and D. Cochran, "Optimal allocation of interconnecting links in cyber-physical systems: Interdependence, cascading failures, and robustness," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1708–1720, 2012.

[2] V. Rosato, L. Issacharoff, F. Tiriticco, S. Meloni, S. Porcellinis, and R. Setola, "Modelling interdependent infrastructures using interacting dynamical models," *Int. J. Critical Infrastructures*, vol. 4, no. 1-2, pp. 63–79, 2008.

[3] C.-G. Gu, S.-R. Zou, X.-L. Xu, Y.-Q. Qu, Y.-M. Jiang, H.-K. Liu, T. Zhou *et al.*, "Onset of cooperation between layered networks," *Physical Review E*, vol. 84, no. 2, p. 026101, 2011.

[4] M. Parandehgheibi and E. Modiano, "Robustness of interdependent networks: The case of communication networks and the power grid," in *Proc. IEEE GLOBECOM*, Atlanta, 2013, pp. 2164–2169.

[5] M. Parandehgheibi, K. Turitsyn, and E. Modiano, "Modeling the impact of communication loss on the power grid under emergency control," in *Proc. IEEE SmartGridComm*, Miami, 2015, pp. 356–361.

[6] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, "Catastrophic cascade of failures in interdependent networks," *Nature*, vol. 464, no. 7291, pp. 1025–1028, 2010.

[7] J. Shao, S. V. Buldyrev, S. Havlin, and H. E. Stanley, "Cascade of failures in coupled network systems with multiple support-dependence relations," *Physical Review E*, vol. 83, no. 3, p. 036116, 2011.

[8] S. Yang, S. Trajanovski, and F. A. Kuipers, "Shortest paths in networks with correlated link weights," in *Proc. IEEE Computer Communications Workshops (INFOCOM WKSHPS)*, Hong Kong, 2015, pp. 660–665.

[9] D. Xu, Y. Xiong, C. Qiao, and G. Li, "Failure protection in layered networks with shared risk link groups," *IEEE network*, vol. 18, no. 3, pp. 36–41, 2004.

[10] D. Coudert, P. Datta, S. Pérennes, H. Rivano, and M.-E. Voge, "Shared risk resource group complexity and approximability issues," *Parallel Processing Letters*, vol. 17, no. 02, pp. 169–184, 2007.

[11] J. Q. Hu, "Diverse routing in optical mesh networks," *IEEE Transactions on Communications*, vol. 51, no. 3, pp. 489–494, 2003.

[12] S. Yuan, S. Varma, and J. P. Jue, "Minimum-color path problems for reliability in mesh networks," in *Proc. INFOCOM 2005*, Miami, 2005, pp. 2658–2669.

[13] H.-W. Lee, E. Modiano, and K. Lee, "Diverse routing in networks with probabilistic failures," *IEEE/ACM Transactions on Networking*, vol. 18, no. 6, pp. 1895–1907, 2010.

[14] M. F. Habib, M. Tornatore, F. Dikbiyik, and B. Mukherjee, "Disaster survivability in optical communication networks," *Computer Communications*, vol. 36, no. 6, pp. 630–644, 2013.

[15] W. Cui, I. Stoica, and R. H. Katz, "Backup path allocation based on a correlated link failure probability model in overlay networks," in *Proc. IEEE Int. Conf. Network Protocols*, Paris, 2002, pp. 236–245.

[16] Q. Zheng, G. Cao, T. F. La Porta, and A. Swami, "Cross-layer approach for minimizing routing disruption in ip networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1659–1669, 2014.

[17] C. Tang and P. K. McKinley, "Improving multipath reliability in topology-aware overlay networks," in *Proc. IEEE International Conference on Distributed Computing Systems Workshops*, Columbus, Ohio, 2005, pp. 82–88.

[18] T. Fei, S. Tao, L. Gao, and R. Guerin, "How to select a good alternate path in large peer-to-peer systems?" in *Proc. INFOCOM 2006*, Barcelona, April 2006, pp. 1–13.

[19] J. Zhang and E. Modiano, "Robust routing in interdependent networks," Tech. Rep., 2016. [Online]. Available: https://www.dropbox.com/s/yzd0ezi0kijnfu7/routing.pdf?dl=0

[20] R. M. Karp, M. Luby, and N. Madras, "Monte-carlo approximation algorithms for enumeration problems," *Journal of algorithms*, vol. 10, no. 3, pp. 429–448, 1989.

[21] W. Gatterbauer and D. Suciu, "Oblivious bounds on the probability of boolean functions," *ACM Transactions on Database Systems (TODS)*, vol. 39, no. 1, p. 5, 2014.

[22] A. P. Punnen, "A linear time algorithm for the maximum capacity path problem," *European J. Oper. Res.*, vol. 53, no. 3, pp. 402–404, 1991.

[23] J. Edmonds and R. M. Karp, "Theoretical improvements in algorithmic efficiency for network flow problems," *Journal of the ACM (JACM)*, vol. 19, no. 2, pp. 248–264, 1972.

[24] XO Communications, "Network map." [Online]. Available: http://www.xo.com/about/network/Pages/maps.aspx