

Reliability Certification of Supply Chain Networks Under Uncertain Failures and Demand

Ashish Chandra^{1*} and Jong Gwang Kim²

^{1*}Department of Management, College of Business, Illinois State University, Normal, IL 61761.

²Department of Economics, Finance and Quantitative Analysis, Coles College of Business, Kennesaw State University, Kennesaw, GA 30144.

*Corresponding author(s). E-mail(s): achand6@ilstu.edu;
Contributing authors: jkim311@kennesaw.edu;

Abstract

In the dynamic and uncertain landscape of modern supply chains, a significant challenge for supply chain network (SCN) architects is ensuring that their designs consistently deliver reliable performance amidst variable traffic demands and unpredictable failures. This assurance is particularly difficult due to the exponential and potentially uncountable set of scenarios that must be considered for evaluation. In response to this challenge, we propose an optimization-theoretic framework aimed at assessing the worst-case performance of supply chain networks by incorporating flexible routing strategies. By modeling network failures and uncertain demand requirements, our methodology tackles the inherent intractability of these certification problems and provides valid upper bounds for the worst-case link utilization and the total unmet demand, essential metrics for maintaining robust network operations. We leverage optimization and convexification techniques to optimize the SCN design under disruptions and demand variation. Through rigorous evaluations on various supply chain network topologies, our results demonstrate the efficacy and promise of our approach. Our framework not only provides a systematic way to certify network performance but also serves as a tool for supply chain architects to enhance resilience against unforeseen disruptions. Ultimately, our research contributes to the development of more reliable and resilient supply chains in an increasingly volatile global environment.

Keywords: Supply Chain Network Resilience, Robust Network Design, Robust Optimization, Reformulation Linearization Technique

1 Introduction

The efficient functioning of contemporary businesses hinges on the robustness of supply chain networks (SCNs), which facilitate the seamless flow of goods and services from suppliers to end consumers [1]. Designing such networks to ensure consistent performance at an acceptable cost is paramount. However, this task is fraught with challenges due to planned maintenance and unplanned events, which can disrupt the flow of goods [2–4]. Addressing these challenges necessitates robust strategies to maintain continuity and efficiency, making the reliability certification of SCNs under uncertain conditions a critical area of research. Supply chain reliability is defined as the network’s ability to consistently meet demand requirements despite disruptions. This reliability is closely tied to the concept of resilience, which refers to the capacity of the supply chain to recover from unforeseen disruptions. Comprehensive reviews on this topic highlight the importance of resilience for maintaining supply chain reliability [5–8]. As SCNs become increasingly global and complex [9], the potential for disruptions escalates, demanding robust approaches to reliability certification.

The COVID-19 pandemic starkly highlighted vulnerabilities within global supply chains. The pandemic-induced disruptions affected production and distribution channels worldwide, underscoring the need for SCNs to be both resilient and adaptable to rapidly changing circumstances [10–12]. Consequently, developing methodologies for certifying the reliability of SCNs to ensure they can withstand and adapt to both predictable and unpredictable disruptions has become urgent.

Traditional supply chain management approaches often emphasize cost minimization and efficiency optimization under stable conditions. However, the dynamic nature of SCNs necessitates models that account for variability and uncertainty. Additionally, there is a growing recognition of the trade-offs between different supply chain objectives, such as sustainability and resilience. While environmental sustainability is crucial, it must not compromise the reliability and resilience of the supply chain [13]. Robust optimization has emerged as a powerful approach for addressing these challenges. For instance, Bertsimas and Thiele [14] discussed inventory control problems using cardinality-constrained uncertainty sets. Qualitative robust strategies for mitigating supply chain disruptions were explored by Tang [15], while You and Grossmann [16] integrated inventory and safety stock considerations under demand uncertainty in their supply chain design models. Peng et al. [17] addressed single-link failures in network disruptions, and Qiu and Wang [18] developed a robust optimization model for designing a three-echelon SCN under both demand uncertainty and supply disruptions.

The complexities of ensuring SCN reliability are not unique to supply chains but are also encountered in other domains such as electrical power networks [19], telecommunication networks [20, 21], and medical services [22]. Recent advancements in simulation and deep learning have further enhanced the study of network reliability problems [23, 24]. Validating that an SCN can cope with a range of demand requirements and failure scenarios is challenging due to the vast number of possible scenarios. For example, verifying that an SCN with links connecting suppliers to plants, plants to warehouses, and warehouses to retailers can satisfy all demand requirements

amidst various link failures involves considering an exponentially large number of failure scenarios. This complexity is further compounded by the non-enumerable nature of uncertain demand requirements.

Robust optimization techniques [25, 26] provide a means to estimate the worst-case SCN performance across multiple scenarios. These bounds are particularly useful when the network can adapt by re-routing flows as demand requirements change or failures occur. However, certifying SCN design today often relies on extensive simulations, which can be time-consuming and may not guarantee provable performance bounds.

Contribution of the paper: This paper proposes a formal framework for providing performance bounds on SCN design across a set of scenarios involving demand requirements and failures. By leveraging cutting-edge techniques in non-linear optimization [27], we address the NP-Hard problems associated with flexible routing strategies. These techniques, notable for their generality, apply to a wide range of SCN performance metrics and certification problems. Our framework not only provides valid upper bounds for certification problems but also demonstrates tight bounds in practical settings, such as when demand requirements are expressed as a convex combination of known historical demands [28]. While our primary focus is on certifying SCNs, the framework also facilitates the synthesis of SCN designs with performance guarantees under uncertainty. We illustrate this by showing how our approach can aid in augmenting link capacities while ensuring acceptable network performance. We validate our approach using multiple SCN topologies, demonstrating how the framework aids in identifying critical failure scenarios and determining optimal strategies for augmenting link capacities to handle failures.

In conclusion, this paper addresses the urgent need for a systematic approach to the reliability certification of SCNs under uncertain failures and demands. By integrating robust optimization techniques and resilience principles, we provide a comprehensive framework that helps businesses navigate the complexities of modern supply chain management, ensuring consistent performance and resilience in the face of uncertainty.

2 Motivation

A supply chain network design consists of (i) constant parameters like the network topology itself, which cannot be changed (or is expensive to change) across uncertain network disruptions; and (ii) adaptive parameters, like the traffic/flow routed on the links of the SCN, that may be chosen flexibly depending on the state of the SCN. Our objective in this paper is to ensure that the chosen constant parameter i.e., the SCN topology performs acceptably across a set of uncertain demand requirements and network failures. Our approach is akin to robust optimization. In conventional robust optimization, input parameters are defined within an uncertainty set, and the goal is to minimize the objective function for any possible parameter within this set [25, 26]. Additionally, recourse actions can be taken based on specific parameter values. In the context of supply chain networks, a common recourse action is to reroute flows to manage disruptions and uncertain demand effectively. A typical set of failure scenarios to consider is all simultaneous failure of $\mathcal{F}$ links [29, 30]. Demand requirements can be outlined in various ways. A typical method is to provide a set of historical demand

data and ensure that all demands projected by standard prediction models are taken into account [16, 31–33].

3 Robust Certification Problem

Let $\mathcal{X}$ denote the uncertainty set of demands or failures over which a given SCN design must be certified. The design includes all parameters that must remain constant/invariant with changes in demands and the occurrence of failures. The SCNs respond to failures, and changes in demand by rerouting flows in the best possible way to minimize/maximize a given objective function. Let y denote the parameters determined by the SCN when adapting to a scenario $x \in \mathcal{X}$. Formally, the SCN certification problem may be written as:

$$g^* = \max_{x \in \mathcal{X}} \min_{y \in Y(x)} g(x, y) \quad (1)$$

The inner minimization captures that for any given $x \in \mathcal{X}$, the SCN determines y in a manner that minimizes an objective function $g(x, y)$ from a set of permissible strategies $Y(x)$. The outer maximization robustly captures the worst-case performance across the set of scenarios in $\mathcal{X}$, assuming that the network adapts in the best possible manner for each x . In our work, we focus on minimizing: (i) link utilization of the most congested link in the SCN (referred to as the maximum link utilization hereafter), and (ii) total unmet demand, as described in Section 3.1. We refer to (1) as the robust certification problem since it can be used to verify if a chosen SCN design meets a desired performance objective threshold. For instance, when applied to a SCN design, if $g(x, y)$ models the maximum link utilization in the SCN, then $g^* > 1$ indicates that the SCN is not sufficiently provisioned to handle all failure scenarios and changing demands of interest.

Given scenario $\bar{x} \in \mathcal{X}$, the inner minimization problem $\min_{y \in Y(\bar{x})} g(\bar{x}, y)$ is generally easy to solve (a linear program) since the network must quickly compute y to adapt to any network state resulting from the uncertainty set. The robust certification problem is however challenging since $\mathcal{X}$ could be a continuous set or may contain exponentially many scenarios.

3.1 Robust certification for Supply chain networks

We consider standard supply chain networks with four levels: suppliers, plants, warehouses, and retailers. These are also commonly referred to as four-echelon supply chain networks [34–40]. We assume that the demands only occur at the retailers and that there are no fixed costs at any of the facilities. These assumptions are standard in the literature, see for reference [29, 41].

We denote a given SCN as a graph $G(V, E)$, where V is the set of nodes, consisting of supplier nodes (S), plant nodes (P), warehouse nodes (W), and retailer nodes (R) such that $V = S \cup P \cup W \cup R$. The set of edges is represented by E , such that $E = E_{SP} \cup E_{PW} \cup E_{WR}$, where E_{SP} represents the set of directed edges connecting the suppliers to the plants, E_{PW} is the set of directed edges connecting the plants to the warehouses, and E_{WR} represents the set of directed edges connecting the warehouses to the retailers. The known link capacity for $\langle i, j \rangle \in E$ is given by $c_{ij} \geq 0$. The

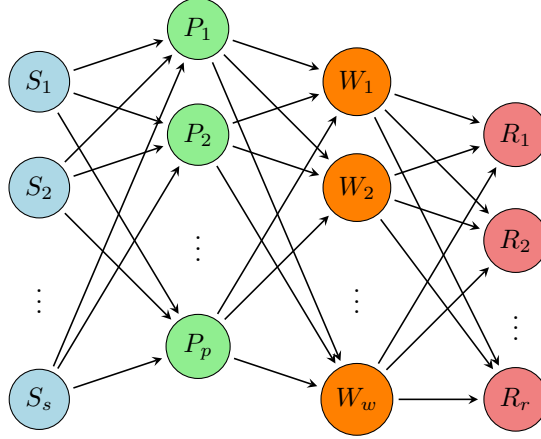


Fig. 1: Standard four echelon supply chain network

flows along all the links in the SCN are the decision variables and they are subject to the maximum flow capacity constraints which we discuss in detail in the subsequent sections (see Table 1 for the notational summary). Figure 1 shows an example of the typical supply chain network we consider for our work. We next relate the general robust certification formulation as described in (1) to our evaluation cases, where our objective is to:

- A1** *certify a SCN design for a performance metric amidst uncertain failures.*
- A2** *certify a SCN design for a performance metric amidst uncertain demands.*

Our work illustrates the generality of our framework in terms of its ability to certify the SCN design across various SCN performance metrics, amidst failures and uncertain demand requirements (discrete and continuous uncertainty sets). For our evaluation cases, we will in particular be focusing on the SCN performance metrics **B1** and **B2**. However, our framework applies to a wider range of applications including simultaneously varying demand requirements and allowing failures, other SCN metrics, and other uncertainty sets (Section 4.4).

- B1** *Link utilization of the most congested link (U).*
- B2** *Total unsatisfied demand across all the retailer (TLD).*

We use the notation $x = (x^f, x^d) \in \mathcal{X}$, where x^f denotes a failure scenario and x^d denotes a particular demand scenario. For **A1**, we assume that the retailer demand requirements are known, and the certification needs to be done amidst uncertain failures $x^f \in \mathcal{X}$, whereas for **A2**, we assume no failures occur in the SCN, and the certification is to be done solely amidst uncertain demand requirements $x^d \in \mathcal{X}$. We will be dropping the superscripts in x^f and x^d when the context is clear. Next, in Sections 3.1.1 and 3.1.2, we discuss the SCN performance metrics **B1** and **B2**. We also analyze how these metrics govern the inner minimization problem in (1) for our evaluation Cases **A1** and **A2**.

Notation	Meaning
S, P, W, R	Known set of supplier, plant, warehouse, and retailer nodes
V	Set of all nodes in the SCN. $V = S \cup P \cup W \cup R$
$s \in S$	Supplier s (s - Index for suppliers)
$p \in P$	Plant p (p - Index for plants)
$w \in W$	Warehouse w (w - Index for warehouses)
$r \in R$	Retailer r (r - Index for retailers)
E_{SP}	Known set of edges from supplier nodes (S) to plant nodes (P)
E_{PW}	Known set of edges from plant nodes (P) to warehouse nodes (W)
E_{WR}	Known set of edges from warehouse nodes (W) to retailer nodes (R)
E	Set of all edges in the SCN. $E = E_{SP} \cup E_{PW} \cup E_{WR}$
c_{ij}	Known maximum flow capacity of the link $\langle i, j \rangle \in E$
D_r	Known demand requirement of retailer $r \in R$
Decision variables	
y_{ij}	Amount of flow to be routed on link $\langle i, j \rangle \in E$
U	Link utilization of the most congested link in the SCN
d_r	Demand out of D_r which is left unsatisfied for retailer $r \in R$

Table 1: List of notations

3.1.1 Supply chain networks under uncertain failures

In this section, we formulate the inner minimization problem in (1), for the Case A1, where a given SCN is exposed to uncertain failures, and we wish to certify the SCN design based on performance metrics B1 and B2. Here, the outer maximization in (1) occurs over $\mathcal{X}$, which is a set of uncertain failures. Let $x^f \in \mathcal{X}$ represent a failure scenario, such that for all links $\langle i, j \rangle$ in the network, we let x_{ij}^f be a binary variable that is 1 if link $\langle i, j \rangle$ (set of links) has failed under scenario x^f , and 0 otherwise. As $x^f \in \mathcal{X}$ captures the functional state of links in the network, hence hereafter, for Case A1, we also refer to (i) x^f as a *link failure scenario*, and (ii) $\mathcal{X}$ as the *set of uncertain link failures*. Under this certification case since we do not consider variable retailer demands, so we let D_r denote the known demand requirement of retailer $r \in R$. We let y denote the quantity determined by the SCN when adapting to a scenario x^f . For every link $\langle i, j \rangle \in E$, we let y_{ij} denote the total flow routed on link $\langle i, j \rangle$, then the link utilization of the most congested link in the SCN i.e., metric B1 is defined as follows:

$$U = \max_{\langle i, j \rangle \in E} U_{ij}, \text{ where } U_{ij} = \frac{y_{ij}}{c_{ij}} \text{ is the utilization of link } \langle i, j \rangle \in E. \quad (2)$$

Leveraging the definition of y , we formulate Constraints (3a) and (3b) to model the flow balance for each plant node $p \in P$, and each warehouse node $w \in W$ respectively. In particular, they model the fact that the total flow arriving into a node i is equal to the total flow leaving i , where $i \in P \cup W$.

$$\sum_{s \in S: \langle s, p \rangle \in E} y_{sp} = \sum_{w \in W: \langle p, w \rangle \in E} y_{pw} \quad \forall p \in P \quad (3a)$$

$$\sum_{p \in P: \langle p, w \rangle \in E} y_{pw} = \sum_{r \in R: \langle w, r \rangle \in E} y_{wr} \quad \forall w \in W \quad (3b)$$

For a given x^f , $M_f(\cdot)$ in (4) seeks to minimize the metric **B1** as defined in (2). Constraints in (4b) ensure that (i) the utilization of the link $\langle i, j \rangle$ is at most U for all the non-failed links; and (ii) none of the failed links carry any flow. Constraints in (4c) ensure that the net inflow to retailer $r \in R$ is equal to its demand requirement D_r .

$$M_f(x^f): \min_{y, U} U \quad (4a)$$

$$y_{ij} \leq U(1 - x_{ij}^f)c_{ij} \quad \forall \langle i, j \rangle \in E \quad (4b)$$

$$\sum_{w \in W: \langle w, r \rangle \in E} y_{wr} = D_r \quad \forall r \in R \quad (4c)$$

$$U, y_{ij} \geq 0 \quad \forall \langle i, j \rangle \in E \quad (4d)$$

(3a), (3b).

We now discuss our second performance metric **B2**, which captures the total unmet demand across all retailers (TLD). For a retailer $r \in R$, we let d_r denote the non-negative demand out of D_r that is unmet, then TLD is computed as:

$$\text{TLD} = \sum_{r \in R} d_r \text{ where } d_r \text{ is as defined in (6c).} \quad (5)$$

For a given x^f , $L_f(\cdot)$ in (6) seeks to minimize the metric **B2** as defined in (5), ensuring that the utilization of all the links in the network is almost unity. We model this restriction on the link utilization using Constraint (6b), which is obtained by fixing $U = 1$ in (4b). In Constraint (6c), we model the unmet demand for retailer r .

$$L_f(x^f): \min_{y, d} \sum_{r \in R} d_r \quad (6a)$$

$$y_{ij} \leq (1 - x_{ij}^f)c_{ij} \quad \forall \langle i, j \rangle \in E \quad (6b)$$

$$d_r = D_r - \sum_{w \in W: \langle w, r \rangle \in E} y_{wr} \quad \forall r \in R \quad (6c)$$

$$d_r, y_{ij} \geq 0 \quad \forall r \in R, \forall \langle i, j \rangle \in E \quad (6d)$$

(3a), (3b).

If $\mathcal{X}$ represents the set of uncertain link failures following Case **A1**, then the SCN design certification problems for metrics **B1** and **B2** seek to solve $\max_{x^f \in \mathcal{X}} M_f(x^f)$ and $\max_{x^f \in \mathcal{X}} L_f(x^f)$ respectively.

3.1.2 Supply chain networks under uncertain demands

In this section, we formulate the inner minimization problem in (1), for Case **A2**, where a given SCN is required to satisfy uncertain retailer demand requirements, and

we wish to certify the SCN design based on performance metrics **B1** and **B2**. Here, the outer maximization in (1) occurs over $\mathcal{X}$, which is the set of uncertain retailer demand requirements. Let $x^d \in \mathcal{X}$ be a demand scenario, such that for all $r \in R$, we let x_r^d denote the demand requirement at retailer node r . Under this certification case, we assume no failures occur in the SCN. For a given demand scenario x^d , we let $M_d(x^d)$ represent the formulation seeking to minimize the metric **B1** as defined in (2). Similarly, we let $L_d(x^d)$ represent the formulation seeking to minimize **B2** as defined in (5), and also restricting the utilization of all links in the SCN to be almost unity. Then, $M_d(x^d)$ and $L_d(x^d)$ can be obtained from $M_f(x^f)$ and $L_f(x^f)$ respectively by: (i) replacing D_r with x_r^d for all $r \in R$, and (ii) enforcing $x_{ij}^f = 0$ for all $\langle i, j \rangle \in E$, as we are only considering uncertain demands. Below we provide the formulations of $M_d(x^d)$ and $L_d(x^d)$ for reference.

$$M_d(x^d) : \min_{y, U} \left\{ U \mid y_{ij} \leq U c_{ij} \quad \forall \langle i, j \rangle \in E, \quad (4d), \quad (3a), \quad (3b) \right. \\ \left. \sum_{w \in W: \langle w, r \rangle \in E} y_{wr} = x_r^d \quad \forall r \in R \right\}. \quad (7)$$

$$L_d(x^d) : \min_{y, d} \left\{ \sum_{r \in R} d_r \mid y_{ij} \leq c_{ij} \quad \forall \langle i, j \rangle \in E, \quad (6d), \quad (3a), \quad (3b) \right. \\ \left. d_r = x_r^d - \sum_{w \in W: \langle w, r \rangle \in E} y_{wr} \quad \forall r \in R \right\}. \quad (8)$$

If $\mathcal{X}$ represents the set of uncertain retailer demand requirements following Case **A2**, then the network design certification problems for the metrics **B1** and **B2** seek to solve $\max_{x^d \in \mathcal{X}} M_d(x^d)$ and $\max_{x^d \in \mathcal{X}} L_d(x^d)$ respectively.

3.2 Nonlinear Formulation

The robust certification problem in (1) has been represented in a form referred to as a two-stage formulation (see e.g., [26, 42–44]). Here, the optimal second-stage variables ($y \in Y(x)$) depend on the first-stage variables ($x \in \mathcal{X}$). We simplify this problem by re-expressing it as a single-stage problem, where all the variables are determined simultaneously. In many robust certification problems, including ours, given $\bar{x} \in \mathcal{X}$, the inner optimization problem $\min_{y \in Y(\bar{x})} g(\bar{x}, y)$ is a linear program in variables $y \in Y(\bar{x})$ for a fixed $\bar{x}$, as is seen in formulations $M_f(\cdot)$, $L_f(\cdot)$, $M_d(\cdot)$, and $L_d(\cdot)$ from Sections 3.1.1 and 3.1.2. It is well known that every LP (referred to as a primal form) involving a minimization objective may be converted into an equivalent maximization LP (referred to as a dual form) which achieves the same objective (assuming the dual is feasible) [45]. The robust certification problem in (1), can then be expressed as a single-stage formulation by, first rewriting $\min_{y \in Y(x)} g(x, y)$ as an equivalent maximization problem using LP duality for a given x , and then adding the constraints $x \in \mathcal{X}$ to the dual form to capture the uncertainty set of interest. We leverage the dual

formulations of $M_f(\cdot)$, $L_f(\cdot)$, $M_d(\cdot)$, and $L_d(\cdot)$ to rewrite their corresponding SCN certification problems (in particular $\max_{x^f \in \mathcal{X}} M_f(x^f)$, $\max_{x^f \in \mathcal{X}} L_f(x^f)$, $\max_{x^d \in \mathcal{X}} M_d(x^d)$, and $\max_{x^d \in \mathcal{X}} L_d(x^d)$) as a single-stage formulation in Sections 3.2.1 and 3.2.2.

3.2.1 1-stage formulation: Certification under uncertain failures

In this section, we construct the single-stage formulations for solving the two-stage SCN certification problems $\max_{x^f \in \mathcal{X}} M_f(x^f)$ and $\max_{x^f \in \mathcal{X}} L_f(x^f)$. For a fixed $x^f \in \mathcal{X}$, we first dualize $M_f(x^f)$ and $L_f(x^f)$ to obtain their corresponding dual maximization problems [45, 46]. We then add constraints defining $\mathcal{X}$ to the obtained dual formulations to capture the uncertainty set under consideration. In particular, we obtain $\mathcal{M}_f$ defined in (9) and $\mathcal{L}_f$ defined in (10) as the single-stage formulations for solving $\max_{x^f \in \mathcal{X}} M_f(x^f)$ and $\max_{x^f \in \mathcal{X}} L_f(x^f)$ respectively.

$$\mathcal{M}_f : \max_{\nu, x^f} \sum_{r \in R} D_r \lambda_r \quad (9a)$$

$$\sum_{\langle i, j \rangle \in E} \alpha_{ij} (1 - x_{ij}^f) c_{ij} \leq 1 \quad (9b)$$

$$\gamma_p \leq \alpha_{sp} \quad \forall \langle s, p \rangle \in E_{SP} \quad (9c)$$

$$\theta_w - \gamma_p \leq \alpha_{pw} \quad \forall \langle p, w \rangle \in E_{PW} \quad (9d)$$

$$\lambda_r - \theta_w \leq \alpha_{wr} \quad \forall \langle w, r \rangle \in E_{WR} \quad (9e)$$

$$\alpha_{ij} \geq 0 \quad \forall \langle i, j \rangle \in E \quad (9f)$$

$$x^f \in \mathcal{X}. \quad (9g)$$

The maximization in $\mathcal{M}_f$ and $\mathcal{L}_f$ happens over the variables $(x^f \in \mathcal{X}, \nu)$, where $\mathcal{X}$ is the failure uncertainty set and $\nu = (\alpha, \gamma, \lambda, \theta)$ are the dual variables obtained post primal dualization. Here, we use α , γ , λ , and θ to represent the set of dual variables $\{\alpha_{ij}\}_{\langle i, j \rangle \in E}$, $\{\gamma_p\}_{p \in P}$, $\{\lambda_r\}_{r \in R}$, and $\{\theta_w\}_{w \in W}$ respectively. We further notice that in the single-stage formulations, the inner minimization variables (y, U, d) from (4) and (6) are replaced by the dual variables $\nu = (\alpha, \gamma, \lambda, \theta)$. Moreover, x^f is now a variable since the certification occurs across all failure scenarios in the uncertainty set $\mathcal{X}$.

$$\mathcal{L}_f : \max_{\nu, x^f} \sum_{r \in R} D_r \lambda_r - \sum_{\langle i, j \rangle \in E} \alpha_{ij} c_{ij} (1 - x_{ij}^f) \quad (10a)$$

$$\lambda_r \leq 1 \quad \forall r \in R \quad (10b)$$

$$(9c - 9g).$$

Proposition 1. *If $M_f(\cdot)$, $L_f(\cdot)$, $\mathcal{M}_f$, and $\mathcal{L}_f$ are as defined in (4), (6), (9), and (10) respectively, then the optimal values of $\mathcal{M}_f$ and $\mathcal{L}_f$ are same as that obtained by solving $\max_{x^f \in \mathcal{X}} M_f(x^f)$ and $\max_{x^f \in \mathcal{X}} L_f(x^f)$ respectively.*

Proof. It is well known that every minimization-linear program (referred to as the primal problem) can be equivalently converted to a maximization-linear program (referred to as its dual problem). Moreover, if the dual problem is feasible, then both

problems achieve the same objective value [45]. For any x^f , clearly $(\alpha, \gamma, \theta, \lambda) = (0, 0, 0, 0)$ is a feasible solution to the dual problems of $M_f(x^f)$ and $L_f(x^f)$, see (9), (10) for reference. Hence, the optimal values of $\mathcal{M}_f$ and $\mathcal{L}_f$ equals the values obtained by solving $\max_{x^f \in \mathcal{X}} M_f(x^f)$ and $\max_{x^f \in \mathcal{X}} L_f(x^f)$ respectively. $\square$

3.2.2 1-stage formulation: Certification under uncertain demands

In this section, we construct the single-stage formulations for solving the two-stage SCN certification problems $\max_{x^d \in \mathcal{X}} M_d(x^d)$ and $\max_{x^d \in \mathcal{X}} L_d(x^d)$. For a fixed demand scenario x^d , we first dualize $M_d(x^d)$ and $L_d(x^d)$, then add the constraints defining the demand uncertainty set $\mathcal{X}$. Let $\mathcal{M}_d$ and $\mathcal{L}_d$ represent the single-stage formulations solving $\max_{x^d \in \mathcal{X}} M_d(x^d)$ and $\max_{x^d \in \mathcal{X}} L_d(x^d)$ respectively. Then, $\mathcal{M}_d$ and $\mathcal{L}_d$ can also be obtained from $\mathcal{M}_f$ and $\mathcal{L}_f$ respectively by: (i) replacing D_r with x_r^d for all $r \in R$, (ii) enforcing $x_{ij}^f = 0$ for all $\langle i, j \rangle \in E$, and (iii) replacing $x^f \in \mathcal{X}$ with the demand uncertainty set $x^d \in \mathcal{X}$. In particular, we obtain $\mathcal{M}_d$ and $\mathcal{L}_d$ as follows:

$$\mathcal{M}_d : \max_{\nu, x^d} \left\{ \sum_{r \in R} x_r^d \lambda_r \mid \sum_{\langle i, j \rangle \in E} \alpha_{ij} c_{ij} \leq 1, (9c) - (9f), x^d \in \mathcal{X} \right\}, \quad (11)$$

$$\mathcal{L}_d : \max_{\nu, x^d} \left\{ \sum_{r \in R} x_r^d \lambda_r - \sum_{\langle i, j \rangle \in E} \alpha_{ij} c_{ij} \mid \lambda_r \leq 1 \forall r \in R, (9c) - (9f), x^d \in \mathcal{X} \right\}, \quad (12)$$

where $\nu = (\alpha, \gamma, \lambda, \theta)$ is the set of dual variables as defined in $\mathcal{M}_f$ and $\mathcal{L}_f$.

Remark 1. If $M_d(\cdot)$, $L_d(\cdot)$, $\mathcal{M}_d$, and $\mathcal{L}_d$ are as defined in (7), (8), (11), and (12) respectively, then the optimal values of $\mathcal{M}_d$ and $\mathcal{L}_d$ are same as that obtained by solving $\max_{x^d \in \mathcal{X}} M_d(x^d)$ and $\max_{x^d \in \mathcal{X}} L_d(x^d)$ respectively. $\square$

Remark 1 follows from Proposition 1, by identifying that for any demand scenario x^d , $(\alpha, \gamma, \theta, \lambda) = (0, 0, 0, 0)$ is a feasible solution to the dual problems of $M_d(x^d)$ and $L_d(x^d)$. Thus, in (9)-(12) we have modeled the single-stage SCN certification problem for performance metrics B1 and B2 amidst uncertain failures (A1) and demands (A2). Solving these formulations is NP-hard, due to the presence of nonlinear terms involving the product of dual variables with variables defining the uncertainty set $\mathcal{X}$.

4 Making SCN Certification Problems Tractable

In Section 3.2, we constructed the single-stage formulation for the SCN certification problem in (1). We observe that these problems are typically intractable. Given the intractable and NP-hard nature of these certification problems, we do not solve them to optimality but rather seek ways to obtain quick upper bounds on the true optimal value of (1). Moreover, since our purpose in certifying a SCN design is to ensure that the design is acceptable, an upper bound that satisfies the design criteria is sufficient. We will be leveraging non-linear programming techniques to come up with such upper bounds. Next, in Section 4.1, we introduce one such upper bounding approach. We then discuss how it applies to our SCN certification cases involving uncertain failures (A1) and demands (A2) in Sections 4.2 and 4.3 respectively.

4.1 Relaxing robust certification problem

Our approach relaxes the robust certification problem into a tractable linear programming problem. This relaxation then allows us to obtain an upper bound on the worst-case SCN performance across all scenarios in the uncertainty set $\mathcal{X}$. An optimization problem $\mathcal{L}$ is a relaxation of a problem $\mathcal{N}$ if (i) every feasible solution in $\mathcal{N}$ can be mapped to a feasible solution in $\mathcal{L}$, and (ii) the mapped solution's objective value in $\mathcal{N}$ is no better than that of its mapping in $\mathcal{L}$. We exploit the Reformulation-Linearization Technique (RLT) [47], which is a general approach to relax non-linear optimization problems. The technique reformulates the non-linear problem by first adding new constraints obtained by taking products of existing constraints and then linearizing the resulting formulation by replacing monomials with new variables. For instance, let us consider a non-linear optimization problem where the objective is to maximize $xy + x - y$ over the variables (x, y) , subject to constraints (i) $xy + x \geq 0$, (ii) $8 - x \geq 0$, (iii) $y - 2 \geq 0$, and (iv) $5 - y \geq 0$. The RLT relaxation for the above problem is obtained by first taking the products of pairs of constraints. For example, the product of constraints (ii) and (iii) results in a newly derived constraint $(8 - x)(y - 2) \geq 0$, i.e., $8y - 16 - xy + 2x \geq 0$. The product term xy is then replaced by a new variable z . The objective function and the constraint (i) $xy + x \geq 0$ are also rewritten as $z + x - y$ and $z + x \geq 0$ respectively. The resulting problem is a linear programming problem since it does not have product terms, however, it is a relaxation of the original non-linear problem, in the sense that constraint (e.g., $z = xy$) that must be present to accurately capture the original problem are not included in the new problem.

The formulation obtained by the above explanation represents the first step in the hierarchy of the RLT relaxations and to go higher in the hierarchy, we multiply more than two constraints and linearize the nonlinear terms as discussed above. Going higher in the hierarchy guarantees tighter bounds, but the corresponding LPs generated become very large (more constraints and variables are introduced), see for reference [47, 48]. Thus, we restrict ourselves to the first level of the RLT relaxation. Furthermore, in practice, it is often sufficient to consider a subset of the product of constraints even in the first level of hierarchy, which keeps the complexity of the resulting relaxation problem manageable. In Sections 4.2 and 4.3, we apply the above RLT relaxation technique to our SCN certification cases A1 and A2 respectively.

4.2 Relaxing certification problem under uncertain failures

In this section, we apply the RLT relaxation technique to the single-stage formulations (9) and (10), obtained in Section 3.2.1 for certifying the SCN design against uncertain failures. For concreteness, we consider all failure scenarios involving the simultaneous failure of $\mathcal{F}$ links in the SCN, such that $\mathcal{F} = f_1 + f_2 + f_3$, where f_1 , f_2 , and f_3 are the number of simultaneous links failing from the edge sets E_{SP} , E_{PW} , and E_{WR} respectively (see Table 1 for reference). Similar link failure models are commonly used in practice see for reference [20, 21, 29, 49]. To incorporate this $\mathcal{F}$ -simultaneous link failure model in Section 3.2.1, we define the failure uncertainty set $\mathcal{X}$ for Case A1 as described in (13), where $\mathcal{F} = f_1 + f_2 + f_3$ and $E = E_{SP} \cup E_{PW} \cup E_{WR}$. We will later

discuss how to generalize this failure model in Section 4.4.

$$\mathcal{X} : \left\{ x = \{0, 1\}^{|E|} : \sum_{\langle s, p \rangle \in E_{SP}} x_{sp} = f_1, \sum_{\langle p, w \rangle \in E_{PW}} x_{pw} = f_2, \sum_{\langle w, r \rangle \in E_{WR}} x_{wr} = f_3 \right\}. \quad (13)$$

With our numerical computations, we observed that for $\mathcal{X}$ defined as in (13), a simple first-level RLT relaxation of $\mathcal{M}_f$ and $\mathcal{L}_f$ (9, 10) does not yield a sufficiently tight upper bound to the certification problems $\max_{x \in \mathcal{X}} M_f(x)$ and $\max_{x \in \mathcal{X}} L_f(x)$ respectively. We thus reformulate the certification problems $(\mathcal{M}_f, \mathcal{L}_f)$ and consequently derive constraints for the RLT relaxation, as described below.

Reformulation of the certification problem. We add variables to (4) and (6) in a way that provides the inner minimization problems more flexibility in choosing a solution, without changing the optimum. Adding variables to the primal problem results in additional constraints to the dual, which eventually show up in the single-stage formulations $(\mathcal{M}_f, \mathcal{L}_f)$ and their associated RLT relaxations. These extra constraints and their corresponding RLT constraints help produce tighter upper bounds for the certification problems. In particular, we reformulate $M_f(x^f)$ and $L_f(x^f)$ using the following steps:

- C1** Augment each link $\langle i, j \rangle \in E$ with extra variable slack capacity $a_{ij} \geq 0$. To model this we replace (4b) with $y_{ij} \leq U(1 - x_{ij}^f)c_{ij} + a_{ij} \forall \langle i, j \rangle \in E$ and (6b) with $y_{ij} \leq (1 - x_{ij}^f)c_{ij} + a_{ij} \forall \langle i, j \rangle \in E$. This modification allows up to a_{ij} of the traffic on link $\langle i, j \rangle$ to be sent over an associated virtual link without counting it towards its utilization. To compensate for this, we increase the total traffic that must be routed from any node i to node j by a_{ij} . We model this in **C2**.
- C2** Replace (3a), (3b), (4c), and (6c) with (14a), (14b), (14c), and (14d) respectively.

$$\sum_{s \in S: \langle s, p \rangle \in E} y_{sp} = \sum_{w \in W: \langle p, w \rangle \in E} y_{pw} + \sum_{s \in S: \langle s, p \rangle \in E} a_{sp} \quad \forall p \in P \quad (14a)$$

$$\sum_{p \in P: \langle p, w \rangle \in E} y_{pw} = \sum_{r \in R: \langle w, r \rangle \in E} y_{wr} + \sum_{p \in P: \langle p, w \rangle \in E} a_{pw} \quad \forall w \in W \quad (14b)$$

$$\sum_{w \in W: \langle w, r \rangle \in E} y_{wr} = D_r + \sum_{w \in W: \langle w, r \rangle \in E} a_{wr} \quad \forall r \in R \quad (14c)$$

$$d_r + \sum_{w \in W: \langle w, r \rangle \in E} (y_{wr} - a_{wr}) = D_r \quad \forall r \in R \quad (14d)$$

We will hereafter refer to the reformulations of $M_f(x^f)$ and $L_f(x^f)$ as $\text{Slack-}M_f(x^f)$ and $\text{Slack-}L_f(x^f)$. It can be shown that $\text{Slack-}M_f(x^f)$ and $\text{Slack-}L_f(x^f)$ constructed using **C1** and **C2** obtain the same optimal as the original formulations described in (4) and (6) respectively.

Proposition 2. *For a given failure scenario $x = \{x_{ij}\}_E$, consider $M_f(x)$ and its corresponding reformulation $\text{Slack-}M_f(x)$. Then, given a feasible solution to either of*

the formulations, a feasible solution to the other can be constructed that has the same objective value.

The above result follows from Proposition 4 in [21]. We further note that, given a failure scenario x , Proposition 2, holds for $L_f(x)$ and its corresponding reformulation $\text{Slack-}L_f(x)$. Let the reformulated primal problems $\text{Slack-}M_f(\cdot)$ and $\text{Slack-}L_f(\cdot)$ yield $\text{Slack-}\mathcal{M}_f$ and $\text{Slack-}\mathcal{L}_f$ as their respective reformulated single-stage certification problems, using the dualizing procedure described in Section 3.2.1. Then, $\text{Slack-}\mathcal{M}_f$ and $\text{Slack-}\mathcal{L}_f$ are obtained as $\mathcal{M}_f$ in (9) and $\mathcal{L}_f$ in (10) respectively, with the following additional constraints:

$$\alpha_{sp} - \gamma_p \leq 0 \quad \forall \langle s, p \rangle \in E_{SP}, \quad (15a)$$

$$\alpha_{pw} - \theta_w \leq 0 \quad \forall \langle p, w \rangle \in E_{PW}, \quad (15b)$$

$$\alpha_{wr} - \lambda_r \leq 0 \quad \forall \langle w, r \rangle \in E_{WR}. \quad (15c)$$

Proposition 3. *Slack- $\mathcal{M}_f$ constructed using the reformulation steps described above achieves the same optimal value as the original certification problem $\mathcal{M}_f$ in (9).*

Proof. The result follows since the optimal values achieved by $\text{Slack-}M_f(x)$ and $M_f(x)$ are equal for all x in the failure uncertainty set using Proposition 2, and hence $\max_x \text{Slack-}M_f(x) = \max_x M_f(x)$. This further implies that their corresponding single-stage formulations $\text{Slack-}\mathcal{M}_f$ and $\mathcal{M}_f$ also achieve the same optimal value. $\square$

We further note that Proposition 3 holds for the reformulation $\text{Slack-}\mathcal{L}_f$ and the original certification problem $\mathcal{L}_f$ in (10). For $\mathcal{X}$ defined in (13), $\text{Slack-}\mathcal{M}_f$ and $\text{Slack-}\mathcal{L}_f$ represent the single-stage formulations for certifying a given SCN design under $\mathcal{F}$ simultaneous link failure scenarios for metrics B1 and B2 respectively. We note that the optimal objective values of $\text{Slack-}\mathcal{M}_f$ and $\text{Slack-}\mathcal{L}_f$ are finite only under the condition that the failure of f_1 links in E_{SP} , f_2 links in E_{PW} , and f_3 links in E_{WR} does not result in the disconnection of the given SCN. This is a condition that can be verified in polynomial time [50]. Since these certification problems are intractable, we derive their first-level RLT relaxations. To construct this relaxation for $\text{Slack-}\mathcal{M}_f$, we start with the $\text{Slack-}\mathcal{M}_f$ formulation, then:

D1 First, the binary constraints $x \in \{0, 1\}^{|E|}$ in (13) defining $\mathcal{X}$ in (9g) are linearized to bounding constraints i.e., $x_{ij} \geq 0$ and $1 - x_{ij} \geq 0$ for all $\langle i, j \rangle \in E$. We hereafter let $\mathcal{X}_L$ denote the set obtained from $\mathcal{X}$ after this linearization.

$$\mathcal{X}_L : \left\{ x = [0, 1]^{|E|} : \sum_{\langle s, p \rangle \in E_{SP}} x_{sp} = f_1, \sum_{\langle p, w \rangle \in E_{PW}} x_{pw} = f_2, \sum_{\langle w, r \rangle \in E_{WR}} x_{wr} = f_3 \right\}. \quad (16)$$

D2 Multiply the constraints defining $\text{Slack-}\mathcal{M}_f$ i.e., (9c) - (9f), (15a) - (15c) with constraints defining $\mathcal{X}_L$ in (16).

D3 The constraints obtained by the above multiplications are added to the original optimization formulation. We further observe that these constraints contain nonlinear terms. We then relax these nonlinear product terms by replacing them with new variables as described in Table 2.

To complete the construction of the first-level RLT relaxation of Slack- $\mathcal{M}_f$, in addition to linearizing the nonlinear terms obtained in Step **D3**, we also linearize the product terms $\{\alpha_{ij}x_{ij}^f\}_{(i,j) \in E}$ appearing in (9b). Using a similar approach as listed in **D1** - **D3**, we construct the first-level RLT relaxation for Slack- $\mathcal{L}_f$. For this case, in Step **D2**, we multiply constraints defining Slack- $\mathcal{L}_f$ i.e., (10b), (9c) - (9f), (15a) - (15c) with constraints defining $\mathcal{X}_L$ in (16). Then in addition to **D3**, we linearized the product terms $\{\alpha_{ij}x_{ij}^f\}_{(i,j) \in E}$ appearing in the objective function (10a).

New linearizing variable	Replacing	For all indices
α_{ijkl}^x	$\alpha_{ij} \times x_{kl}$	$\langle i, j \rangle, \langle k, l \rangle \in E$
γ_{pkl}^x	$\gamma_p \times x_{kl}$	$p \in P, \langle k, l \rangle \in E$
θ_{wkl}^x	$\theta_w \times x_{kl}$	$w \in W, \langle k, l \rangle \in E$
λ_{rkl}^x	$\lambda_r \times x_{kl}$	$r \in R, \langle k, l \rangle \in E$

Table 2: RLT - Variable description for Case **A1**

4.3 Relaxing certification problem under uncertain demands

In this section, we first explore a model for specifying uncertain demands and then apply the RLT relaxation technique to the single-stage formulations (11) and (12), obtained in Section 3.2.2, for certifying the SCN design against uncertain retailer demands. The model we utilize to capture uncertain demands mimics scenarios where demands can be predicted from past demand data. Let $\delta^t = (\delta_{r_1}^t, \delta_{r_2}^t, \dots, \delta_{r_{|R|}}^t)$ for $t \in T$ represent the vector of known historical demand requirement for the retailers at time t . Given a set of historical demand requirements $\{\delta^t\}_{t \in T}$, where T is a set of time instances from the past, it is reasonable to certify the SCN design over the convex hull of $\{\delta^t\}_{t \in T}$ [51]. This ensures the certification of the SCN design across all convex combinations of the past demand predictors. In particular, this is modeled by having $x^d \in \mathcal{X}$ in Section 3.2.2, where $\mathcal{X}$ is defined as in (17). This set definition is effective when the uncertain demand requirements fall into the convex hull. However, if demands fall outside the convex hull, then one way to handle this issue is to expand the convex hull by letting $\{\mu_t\}_{t \in T}$ take values less than 0 or larger than 1.

$$\mathcal{X} : \left\{ x^d = \sum_{t \in T} \mu_t \delta^t \mid \sum_{t \in T} \mu_t = 1, \mu_t \geq 0 \forall t \in T \right\}. \quad (17)$$

With $\mathcal{X}$ defined as in (17), the SCN certification problems $\mathcal{M}_d$ and $\mathcal{L}_d$ in Section 3.2.2 are intractable. This is due to the presence of nonlinear terms in the objective functions i.e., $\sum_{r,t} \lambda_r \mu_t \delta_r^t$. Following the discussion in Section 4.2, from our numerical computations we observed that a first-level RLT relaxation of $\mathcal{M}_d$ (11) and $\mathcal{L}_d$ (12) does not yield a sufficiently tight upper bound to $\max_{x^d \in \mathcal{X}} M_d(x^d)$ and $\max_{x^d \in \mathcal{X}} L_d(x^d)$ respectively. We thus leverage the reformulation process stated in **C1** and **C2** to reformulate $M_d(x^d)$ and $L_d(x^d)$ as follows:

E1 Augment each link $\langle i, j \rangle \in E$ with extra variable slack capacity $a_{ij} \geq 0$. To do this in (7) and (8), for all $\langle i, j \rangle \in E$, replace Constraints $y_{ij} \leq Uc_{ij}$ and $y_{ij} \leq c_{ij}$ with $y_{ij} \leq Uc_{ij} + a_{ij}$ and $y_{ij} \leq c_{ij} + a_{ij}$ respectively.

E2 In (7) and (8), replace Constraints in (3a) and (3b) with (14a) and (14b) respectively. Also, for all $r \in R$, replace the Constraints $\sum_{w \in W: \langle w, r \rangle \in E} y_{wr} = x_r^d$

and $d_r = x_r^d - \sum_{w \in W: \langle w, r \rangle \in E} y_{wr}$ with (14c) and (14d) respectively, where D_r is substituted with x_r^d for all $r \in R$.

We hereafter refer to Slack- $M_d(x^d)$ and Slack- $L_d(x^d)$ to represent the reformulations constructed for $M_d(x^d)$ and $L_d(x^d)$ respectively. Given a demand requirement scenario x^d , it can be shown that $M_d(x^d)$ and Slack- $M_d(x^d)$ as well as $L_d(x^d)$ and Slack- $L_d(x^d)$ achieve the same optimal objective values.

Let the reformulated primal problems Slack- $M_d(\cdot)$ and Slack- $L_d(\cdot)$ yield Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ as their respective reformulated single-stage certifications problems. Then, Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ are obtained as $\mathcal{M}_d$ in (11) and $\mathcal{L}_d$ in (12) respectively, with the additional constraints in (15a) - (15c).

Remark 2. Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ constructed using the reformulation steps described above achieves the same optimal value as the original certification problems $\mathcal{M}_d$ and $\mathcal{L}_d$ respectively. $\square$

For $\mathcal{X}$ defined in (17), Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ represent the single-stage formulations for certifying a given SCN design under uncertain demand for metrics **B1** and **B2** respectively. Following the equivalence stated in Remark 2, we construct the first-level RLT relaxations for Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ that obtains an upper bound to $\max_{x^d \in \mathcal{X}} M_d(x^d)$ and $\max_{x^d \in \mathcal{X}} L_d(x^d)$ respectively. To construct the relaxation for Slack- $\mathcal{M}_d$, we start with Slack- $\mathcal{M}_d$ formulation, then:

F1 Multiply the constraints defining Slack- $\mathcal{M}_d$ in (11) i.e., $1 - \sum_{\langle i, j \rangle \in E} \alpha_{ij} c_{ij} \geq 0$, (9c) - (9f), (15a) - (15c) with constraints defining $\mathcal{X}$ in (17).

F2 The constraints obtained by the above multiplications are added to the original optimization formulation. These constraints contain nonlinear product terms, that are relaxed by introducing new variables as described in Table 3. Finally, we also linearized the product terms appearing in the objective function $\sum_{t, r} \lambda_r \mu^t \delta_r^t$.

Using a similar approach as described in **F1** and **F2**, we construct the first-level RLT relaxation of Slack- $\mathcal{L}_d$. For this, in Step **F1**, we multiply constraints defining Slack- $\mathcal{L}_d$ in (12) i.e., $1 - \lambda_r \geq 0 \forall r \in R$, (9c) - (9f), (15a) - (15c) with constraints defining $\mathcal{X}$ in (17) and then follow it with Step **F2**.

Proposition 4. For the demand uncertainty set $\mathcal{X}$ defined in (17), the first-level RLT relaxations for Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ constructed above are exact i.e., they achieve the same optimal value as obtained by solving Slack- $\mathcal{M}_d$ and Slack- $\mathcal{L}_d$ respectively.

The result follows from [28] and the definition of $\mathcal{X}$ in (17).

4.4 SCN certification: Over more uncertainty sets

In Sections 4.2 and 4.3, we discussed how the robust certification problem can be used to certify a network design across random failures and uncertain demands. We now

New linearizing variable	Replacing	For all indices
α_{ijt}^μ	$\alpha_{ij} \times \mu_t$	$\langle i, j \rangle \in E, t \in T$
γ_{pt}^μ	$\gamma_p \times \mu_t$	$p \in P, t \in T$
θ_{wt}^μ	$\theta_w \times \mu_t$	$w \in W, t \in T$
λ_{rt}^μ	$\lambda_r \times \mu_t$	$r \in R, t \in T$

Table 3: RLT - Variable description for Case **A2**

discuss how our framework can extend to tackle SCN certification problems beyond **A1** and **A2**, to more uncertainty sets.

Certification amidst node failures. We have already considered an uncertain link failure model where a total of $\mathcal{F} = f_1 + f_2 + f_3$ links fail simultaneously (13). Let $\mathcal{G}$, be a set of link groups such that each link group $g \in \mathcal{G}$ is constituent of a set of links that will fail together. For every $g' \in \mathcal{G}$, we introduce a binary variable $x_{g'}$ which indicates if the link group $g' \in \mathcal{G}$ has failed or not. We use this construction to model node failures. If V represents the set of all nodes in the SCN, then for each node $v' \in V$, we define the link group $g_{v'}$, consisting of all the links entering and leaving v' . For instance, Figure 2 highlights the link group associated with node P_2 , referred to as g_{P_2} , consisting of edges leading into and out of P_2 . The SCN certification problem for node failures is obtained by replacing the constraints defining $x^f \in \mathcal{X}$ in Section 3.2.1 with the following constraints:

$$\sum_{v \in V} x_v = \mathcal{N}, \quad x_{ij}^f = 1 - (1 - x_i)(1 - x_j) \quad \forall \langle i, j \rangle \in E, \quad (18)$$

where x_{ij}^f for all $\langle i, j \rangle \in E$ and x_v for all $v \in V$ are binary variables, taking a value of 1 if the corresponding link or node fails and zero otherwise. The first constraint in (18), models that a total of $\mathcal{N}$ nodes have failed in the SCN. The second constraint captures that a link $\langle i, j \rangle$ fails if and only if at least one group among $\{g_i, g_j\}$ that it belongs to fails. To eliminate the product terms in (18), the second constraint can be linearized with the constraints $x_{ij}^f \geq x_i$, $x_{ij}^f \geq x_j$, and $x_{ij}^f \leq x_i + x_j$. An RLT relaxation may now be applied to this formulation as discussed in the previous sections.

Certification amidst uncertain failures and demands simultaneously. We may desire to certify the SCN design across any combination of simultaneous uncertain failures and demands. This can easily be modeled as a single-stage certification problem by taking Slack- $\mathcal{M}_f$ and Slack- $\mathcal{L}_f$ as described in Sections 4.2 and 4.3 respectively, and then replacing the demand parameter D_r with variable x_r^d for all $r \in R$. We further add constraints defining the uncertainty set $x \in \mathcal{X} : \{x = (x^f, x^d)\}$, such that x^f and x^d belong to the sets defined by (13) and (17) respectively. An RLT relaxation similar to the ones described in Sections 4.2 and 4.3 can then be applied to this case.

4.5 Link capacity augmentation for designing robust SCN

In this Section, we discuss how our certification framework can help design robust SCNs. We focus our discussion on Case **A1** i.e., uncertain link failures and metric

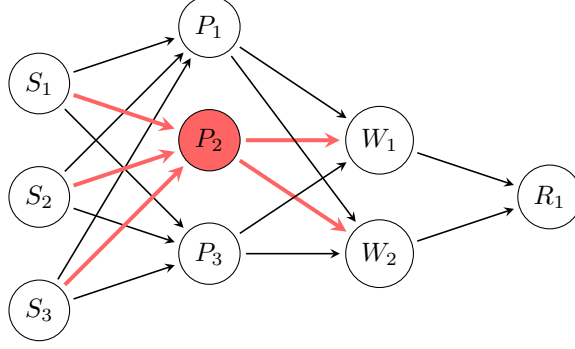


Fig. 2: SCN under node failures - Link group for P_2

B1 i.e., the utilization of the most congested link in the network (U) as defined in (2), but a similar idea would extend to other certification cases and performance metrics as well. Given a SCN, a set of failure scenarios $\mathcal{X}$ satisfying (13), and a desirable network utilization threshold value U' . We consider the problem of adding capacities to existing links in the SCN to ensure all failure scenarios in $\mathcal{X}$ can be handled with a maximum link utilization (U) satisfying $U \leq U'$, while also minimizing the total cost of augmenting capacities. We can extend our robust certification problem $\max_{x^f \in \mathcal{X}} \text{Slack-M}_f(x^f)$ as described in Section 4.2, to model this problem as follows:

$$\min_{\epsilon} \max_{x^f} \min_{y, a} \sum_{\langle i, j \rangle \in E} \rho_{ij} \epsilon_{ij} \quad (19a)$$

$$y_{ij} \leq U'(1 - x_{ij}^f)(c_{ij} + \epsilon_{ij}) + a_{ij} \quad \forall \langle i, j \rangle \in E \quad (19b)$$

$$a_{ij}, \epsilon_{ij}, y_{ij} \geq 0 \quad \forall \langle i, j \rangle \in E \quad (19c)$$

$$x^f \in \mathcal{X} \quad (19d)$$

$$(14a), (14b), (14c),$$

where $\mathcal{X}$ is the set of failure scenarios as defined in (13). For all $\langle i, j \rangle \in E$, ϵ_{ij} represents the variable incremental capacity added to the link $\langle i, j \rangle$ whereas $\rho_{ij} \geq 0$ is the known parameter representing the cost per unit capacity for link $\langle i, j \rangle$. For a fixed ϵ and x^f , we dualize the inner minimization problem in variables (y, a) . This results in (19) being equivalently written as a two-stage (min-max) formulation, whose inner problem has variables $x^f \in \{0, 1\}^{|E|}$. Using the RLT relaxation techniques discussed in Section 4.2, we replace this inner maximization problem with an upper-bounding linear program (LP), which can again be dualized to upper-bound the optimal cost of link augmentation. This provides us with an LP-based approach to conservatively augment capacities, which results in a conservative robust SCN design.

Link capacity augmentation procedure: For a given failure scenario $x^f \in \mathcal{X}$, the

link capacity augmentation problem is solved as the following linear program.

$$\mathcal{A}(x^f) : \min_{y, a, \epsilon} \left\{ \sum_{\langle i, j \rangle \in E} \rho_{ij} \epsilon_{ij} \mid (14a), (14b), (14c), (19b), (19c) \right\}. \quad (20)$$

The above formulation can be extended to every link failure scenario in $\mathcal{X}$ as defined in (13), by replicating the constraints in (20) for every scenario or equivalently by solving $\max_{x^f \in \mathcal{X}} \mathcal{A}(x^f)$. To solve $\max_{x^f \in \mathcal{X}} \mathcal{A}(x^f)$, we first dualize $\mathcal{A}(x^f)$ to obtain $\mathcal{A}_D(x^f)$ as described in (21), and then equivalently solve $\max_{x^f \in \mathcal{X}} \mathcal{A}_D(x^f)$. Since, $\max_{x^f \in \mathcal{X}} \mathcal{A}_D(x^f)$ is a nonlinear optimization problem, with $x^f \in \mathcal{X}$ being a binary variable, an RLT-relaxation for it can be constructed to obtain an upper bound for the total cost of capacity augmentation.

$$\mathcal{A}_D(x^f) : \max_{\alpha, \gamma, \lambda, \theta} \sum_{r \in R} \lambda_r D_r - \sum_{\langle i, j \rangle \in E} U'(1 - x_{ij}^f) c_{ij} \alpha_{ij} \quad (21a)$$

$$\alpha_{ij} U'(1 - x_{ij}^f) \leq \rho_{ij} \quad \forall \langle i, j \rangle \in E \quad (21b)$$

$$\alpha_{ij} \geq 0 \quad \forall \langle i, j \rangle \in E \quad (21c)$$

$$(9c), (9d), (9e), (15a), (15b), (15c).$$

In Section 5.4, we report results on this using various SCN topologies.

5 Evaluation

To demonstrate the effectiveness of our techniques, we perform numerical experiments on four-echelon supply chain networks. Using the network metrics B1 and B2, in Sections 5.1 and 5.3, we solve the SCN certification problems amidst $\mathcal{F}$ -simultaneous link failures (13) and uncertain demands (17) respectively. Further, in Section 5.2 we show that our approach aids in identifying scenarios from the $\mathcal{F}$ -simultaneous link failure set $\mathcal{X}$, that adversely affects the network performance metric under consideration. Then, in Section 5.4, we show that we can augment link capacities to guarantee a given level of SCN performance across all link failure scenarios in $\mathcal{X}$ (13).

We evaluate our work using the SCN topologies listed in Table 4. For each topology T_i , listed under the column labeled “T”, the corresponding entries in the columns labeled |S|, |P|, |W|, and |R| represent the number of supplier nodes, plant nodes, warehouse nodes, and retailer nodes respectively in the SCN. For setting up our testing topologies, we further assume that each directed edge: (i) from all supplier nodes to all plant nodes, (ii) from all plant nodes to all warehouse nodes, and (iii) from all warehouse nodes to all retailer nodes, exists with a probability p , reported in the Column “p”. As stated in Table 1, E_{SP} , E_{PW} , and E_{WR} represent the set of edges connecting the suppliers-plants, plants-warehouses, and warehouses-retailers respectively. Entries corresponding to $|E_{SP}|$, $|E_{PW}|$, and $|E_{WR}|$ in Table 4 represent the cardinality of these edge sets. Further, $|V| = |S| + |P| + |W| + |R|$ and $|E| = |E_{SP}| + |E_{PW}| + |E_{WR}|$, represent the total number of nodes and edges in the SCN topology respectively. For each topology, we assume the link capacities $\{c_{ij}\}_{\langle i, j \rangle \in E}$ to be uniformly generated from the

interval $[10, 50]$. To validate SCNs for performance metrics **B1** and **B2** under uncertain failures (i.e., Case **A1**), we assume the retailer demand requirements $\{D_r\}_{r \in R}$ to be uniformly generated from the interval $[20, 40]$. Then, the total demand requirement across all retailers is obtained as $\sum_{r \in R} D_r$, which we report in Column “T.Dem”.

T	 S 	 P 	 W 	 R 	 E_{SP} , E_{PW} , E_{WR} 	 V , E 	p	T. Dem
T ₁	4	3	2	2	(12, 6, 4)	(11, 22)	1.0	46.252
T ₂	6	5	6	4	(30, 30, 24)	(21, 84)	1.0	117.062
T ₃	7	3	4	7	(21, 12, 28)	(21, 61)	1.0	209.220
T ₄	5	2	3	9	(10, 6, 27)	(19, 43)	1.0	253.199
T ₅	9	10	12	8	(73, 97, 77)	(39, 247)	0.8	257.726
T ₆	10	3	10	15	(17, 19, 101)	(38, 137)	0.7	457.088
T ₇	11	6	9	17	(35, 26, 76)	(43, 137)	0.5	493.250

Table 4: Supply chain network topologies used for evaluation

Implementation: The models were implemented in Python, and all the optimization problems were solved using Gurobi 9.0 [52]. The computations were done on a machine with an Intel Xeon E5-2623 CPU @3.00 GHz.

5.1 Worst-case analysis of network performance across failures

In this section, we discuss the computational results of the SCN certification problem for metrics **B1** (in Table 5) and **B2** (in Table 6) under uncertain link failures as described in Case **A1**. The Column labeled “T (f_1, f_2, f_3)” in Tables 5 and 6 denote the SCN topology and the link failure scenario set $\mathcal{X}$ (as described in (13)) under consideration.

In Table 5, Column labeled “IP”, we report the worst case utilization of the most congested link in the network (U), across all failure scenarios in the set $\mathcal{X}$, by solving the nonlinear integer problem Slack- $\mathcal{M}_f$, which is the single stage formulation for $\max_{x^f \in \mathcal{X}} \text{Slack-M}_f(x^f)$. We describe the construction of Slack- $\mathcal{M}_f$ in Section 4.2. Further, in the Column labeled “RLT”, we report the optimal values for U obtained by solving the first-level RLT relaxation of Slack- $\mathcal{M}_f$ constructed using the Steps **D1-D3**. Testing instances where the optimal U is unbounded are denoted by * (see for instance T₁ (2, 3, 2), T₇ (3, 4, 2) in Table 5). These are the instances, in which the demand requirements at the retailer nodes are not fully satisfied.

In Table 6, the Column labeled “IP”, reports the optimal values for the worst case total lost demand (TLD), across all failure scenarios in the set $\mathcal{X}$, by solving the nonlinear integer problem Slack- $\mathcal{L}_f$, which is the single stage formulation for $\max_{x^f \in \mathcal{X}} \text{Slack-L}_f(x^f)$. We describe the construction of Slack- $\mathcal{L}_f$ in Section 4.2. Further, in the Column labeled “RLT”, we report the optimal values of TLD obtained by solving the first-level RLT relaxation of Slack- $\mathcal{L}_f$ constructed using the Steps **D1-D3**.

Computational time. From our experiments, we observe that the computational time stays stable for solving the first-level RLT relaxation, but explodes for the IP (Slack- $\mathcal{M}_f$, Slack- $\mathcal{L}_f$), as we increase the network size or increase the number of link failures. Since many IP instances, did not finish even after several hours, we set a

10-hour limit for the solver and reported the lower (LB) - upper (UB) bound for the optimal objective value at termination. We also report the optimality gap, computed as $\text{gap} = \frac{\text{UB}-\text{LB}}{\text{UB}} \times 100$ for each of these IP instances, that did not converge within 10 hours of runtime (see e.g., T_5 (4, 3, 4), T_5 (5, 6, 5), T_5 (5, 7, 6), and T_7 (3, 4, 2) in Table 6). For IP instances corresponding to T_5 in Table 5, the solver did not find a valid UB (denoted by $\dagger$) even after 10 hours (the optimality gap hence is undefined, denoted by $\ddagger$), whereas the RLT relaxations were solved in less than a minute.

Computational accuracy. Our experiments confirm that the proposed first-level RLT relaxations provide quick overestimates to the optimal value of $\max_{x^f \in \mathcal{X}} \text{Slack-M}_f(x^f)$ in Table 5 and $\max_{x^f \in \mathcal{X}} \text{Slack-L}_f(x^f)$ in Table 6. Moreover, the results in Tables 5 and 6, can be mutually validated to verify their accuracy and consistency. For e.g., the testing instances in Table 5, with $U^* \leq 1$ ($U^* > 1$), correspond to $\text{TLD}^* = 0$ ($\text{TLD}^* > 0$) in Table 6. This is true since in Table 6 the optimal value of TLD is computed ensuring U is almost one (see Constraint (6b) in $L_f(\cdot)$).

$T(f_1, f_2, f_3)$	IP	T-IP	RLT	T-RLT	T-Dev%	V-Dev%
$T_1(1, 1, 1)$	2.084	0.21	2.084	0.13	38.10	0
$T_1(2, 2, 1)$	2.084	0.91	2.084	0.07	92.31	0
$T_1(2, 3, 2)$	*	*	*	*	*	*
$T_2(1, 2, 1)$	0.371	238.34	0.371	1.96	99.18	0
$T_2(2, 2, 2)$	0.532	21178.92	0.532	3.91	99.98	0
$T_2(1, 2, 3)$	0.728	4002.25	0.728	3.64	99.91	0
$T_3(1, 2, 2)$	1.682	131.31	1.682	2.08	98.42	0
$T_3(1, 2, 3)$	3.592	511.93	3.592	2.04	99.60	0
$T_3(2, 3, 2)$	3.592	5043.29	4.723	2.12	99.96	23.95
$T_4(1, 1, 2)$	2.416	11.64	2.579	0.55	95.27	6.32
$T_4(2, 1, 2)$	2.746	43.50	3.110	0.54	98.76	11.70
$T_4(1, 1, 3)$	*	*	*	*	*	*
$T_5(3, 2, 2)$	(0.237, $\dagger$, $\ddagger$)	36000	0.237	32.22	-	-
$T_5(4, 3, 4)$	(0.444, $\dagger$, $\ddagger$)	36000	0.444	27.37	-	-
$T_5(5, 6, 5)$	(0.917, $\dagger$, $\ddagger$)	36000	0.917	40.56	-	-
$T_5(5, 7, 6)$	(1.920, $\dagger$, $\ddagger$)	36000	1.920	35.07	-	-
$T_6(1, 1, 1)$	1.211	212.36	1.217	61.46	71.06	0.49
$T_6(1, 1, 2)$	1.917	1159.64	1.917	28.98	97.50	0
$T_6(1, 2, 1)$	1.767	862.34	1.767	38.91	95.49	0
$T_6(4, 3, 2)$	*	*	*	*	*	*
$T_7(2, 0, 1)$	1.050	190.05	1.050	21.25	88.82	0
$T_7(1, 1, 1)$	1.566	188.15	1.566	37.08	80.29	0
$T_7(0, 2, 1)$	3.366	190.24	3.366	20.14	89.41	0
$T_7(1, 1, 2)$	*	*	*	*	*	*
$T_7(3, 4, 2)$	*	*	*	*	*	*

Table 5: Solving for $U^* = \text{Slack-M}_f = \max_{x^f \in \mathcal{X}} \text{Slack-M}_f(x^f)$ for $\mathcal{X}$ as in (13)

¹ T-IP: CPU run-time (seconds) for getting the value of U reported in Column “IP”.

² T-RLT: CPU run-time (seconds) for getting the value of U reported in Column “RLT”.

³ T-Dev% = $\frac{\text{T-IP}-\text{T-RLT}}{\text{T-IP}} \times 100$, V-Dev% = $\frac{\text{RLT}-\text{IP}}{\text{RLT}} \times 100$.

⁴ The tuple entries in Column “IP”: (LB, UB, gap).

$T(f_1, f_2, f_3)$	IP	T-IP	RLT	T-RLT	T-Dev%	V-Dev%
$T_1(1, 1, 1)$	12.823	0.17	12.823	0.17	0	0
$T_1(2, 2, 1)$	12.823	0.66	12.823	0.19	71.21	0
$T_1(2, 3, 2)$	46.252	0.65	46.252	0.18	72.31	0
$T_2(1, 2, 1)$	0	8.50	0	8.13	4.35	0
$T_2(2, 2, 2)$	0	8.05	0	7.82	2.86	0
$T_2(1, 2, 3)$	0	9.45	0	8.09	14.39	0
$T_3(1, 2, 2)$	15.231	21.17	15.231	3.13	85.21	0
$T_3(1, 2, 3)$	27.115	22.41	31.479	3.23	85.59	13.86
$T_3(2, 3, 2)$	27.115	84.23	30.958	3.17	96.24	12.41
$T_4(1, 1, 2)$	139.431	2.24	145.053	1.06	52.68	3.88
$T_4(2, 1, 2)$	161.007	2.74	164.231	1.64	40.15	1.96
$T_4(1, 1, 3)$	139.431	2.33	148.577	0.93	60.09	6.16
$T_5(3, 2, 2)$	0	1528.63	0	95.66	93.74	0
$T_5(4, 3, 4)$	(0, 120.793, 100%)	36000	0	52.28	-	-
$T_5(5, 6, 5)$	(0, 172.821, 100%)	36000	0	35.95	-	-
$T_5(5, 7, 6)$	(16.983, 185.501, 90.84%)	36000	16.983	49.73	-	-
$T_6(1, 1, 1)$	79.538	62.07	81.552	29.39	52.65	2.47
$T_6(1, 1, 2)$	79.538	145.23	83.860	57.91	60.13	5.15
$T_6(1, 2, 1)$	118.185	132.85	122.061	27.83	79.05	3.18
$T_6(4, 3, 2)$	251.784	2381.05	380.118	41.17	98.27	33.76
$T_7(2, 0, 1)$	1.660	38.28	1.660	18.15	52.59	0
$T_7(1, 1, 1)$	12.651	55.66	12.651	18.56	66.65	0
$T_7(0, 2, 1)$	27.682	41.28	27.682	13.66	66.91	0
$T_7(1, 1, 2)$	36.995	32.13	36.995	13.23	58.82	0
$T_7(3, 4, 2)$	(86.581, 259.360, 66.62%)	36000	258.071	23.76	-	-

Table 6: Solving for $TLD^* = \text{Slack-}\mathcal{L}_f$ ensuring $U \leq 1$, $\mathcal{X}$ as in (13)

¹ $\text{Slack-}\mathcal{L}_f = \max_{x \in \mathcal{X}} \text{Slack-L}_f(x^f)$

² T-IP: CPU run-time (seconds) for obtaining the value of TLD reported under Column “IP”.

³ T-RLT: CPU run-time (seconds) for obtaining the value of TLD reported under Column “RLT”.

⁴ $T\text{-Dev}\% = \frac{T\text{-IP} - T\text{-RLT}}{T\text{-IP}} \times 100$, $V\text{-Dev}\% = \frac{RLT - IP}{RLT} \times 100$.

⁵ The tuple entries in Column “IP”: (LB, UB, gap).

5.2 Identifying scenarios resulting in poor SCN performance

Consider a SCN and a given acceptable threshold value $\varrho \in \mathbb{R}$ for the network performance metric **B2** i.e., the total lost demand. Let Bad-F be the set as described in (22), where $\mathcal{X}$ as in (13), is the set of link failure scenarios under consideration. Then, Bad-F identifies all failure scenarios from $\mathcal{X}$ which results in a total lost demand value that is higher than ϱ .

$$\text{Bad-F} = \left\{ x \in \mathcal{X} : \max_{x \in \mathcal{X}} \text{Slack-L}_f(x) > \varrho \right\}, \text{ where } \mathcal{X} \text{ is as in (13)}. \quad (22)$$

We can further extend this idea to construct the set of link failure scenarios from $\mathcal{X}$, that result in a high value for the performance metric **B1**. In particular, the set of all failure scenarios from $\mathcal{X}$, that cause the utilization of the most congested link in the

SCN to exceed $\tilde{\varrho} = 1$ i.e., $\{x \in \mathcal{X} : \max_{x \in \mathcal{X}} \text{Slack-M}_f(x) > \tilde{\varrho}\}$. In general, identifying link failure scenarios from the failure uncertainty set $\mathcal{X}$, that results in a poor SCN performance (high value of U or TLD) is an NP-Hard problem. An enumerative search is also intractable, e.g., for T_6 in Table 4, a brute-force listing approach showed that only 0.62% of all $(f_1, f_2, f_3) = (1, 1, 1)$ link failure scenarios from $\mathcal{X}$ defined in (23) exceeded the total lost demand threshold value of 75 units, while only 1.24% scenarios exceeded the total lost demand threshold value of 70 units.

To identify failure scenarios in Bad-F (22), we propose a branch and bound (B-B) algorithm that utilizes our first-level RLT LP-relaxation of the original certification problem, $\max_{x \in \mathcal{X}} \text{Slack-L}_f(x)$. At each exploration step (corresponding to the tree nodes in Figure 3) of the B-B algorithm, we fix the functionality status of a subset of links to one (denoting dead link) or zero (denoting active link) and then solve the first-level RLT-relaxation of the original certification problem. We further note that at the root node i.e., at the initial step in Level 0, none of the link functionalities are fixed. The RLT-relaxation run at a node finds an optimal x -solution (possibly non-binary), satisfying the node's fixed link functionality status that results in the optimal/highest total lost demand (TLD) for the relaxation (see the numeric entries in the tree nodes of Figure 3, denoting the highest TLD value obtained by solving the RLT-relaxation). The link with the highest fractional failure (say $\langle k, l \rangle$) is considered, and the RLT-relaxation is rerun fixing x_{kl} as each of 0 and 1. Branches where the TLD value does not exceed ϱ are pruned (since these are acceptable SCN performances). Of the remaining candidate unexplored nodes, the node with the highest TLD value is selected for exploration. Ties are broken by picking the node at the lowest level in the B-B tree. The process is run until an integral solution for x is found, and the search procedure can be continued to identify multiple integral solutions for x . Figure 3, describes the B-B algorithm for topology T_6 , for identifying link failure scenarios in Bad-F, when $\varrho = 75$ and $\mathcal{X}$ captures all $(f_1, f_2, f_3) = (1, 1, 1)$ link failures defined as follows:

$$\mathcal{X} : \left\{ x = \{0, 1\}^{|E|} : \sum_{\langle s, p \rangle \in E_{SP}} x_{sp} = 1, \sum_{\langle p, w \rangle \in E_{PW}} x_{pw} = 1, \sum_{\langle w, r \rangle \in E_{WR}} x_{wr} = 1 \right\}. \quad (23)$$

The represented state of the B-B tree in Figure 3, reveals that nodes numbered 4, 6, and 10 provide an integral solution for x , resulting in the TLD value being more than $\varrho = 75$ units, which we wished to identify. In addition, since the optimal/highest TLD value at node 7 is less than the threshold value $\varrho = 75$, so we do not explore node 7 further. Lastly, the figure also reveals that nodes 1 and 9 are active for further exploration to find potential failure scenarios resulting in TLD being more than ϱ .

Identification of all integral solutions for x resulting in unsatisfactory SCN performance (e.g., $\text{TLD} > \varrho$, $U > \tilde{\varrho}$), contributes to designing more failure resilient SCNs. Solving $\mathcal{A}(x)$ described in (20), for each of the identified failure scenarios obtained via the B-B algorithm, allows us to augment link capacities in a manner that permits the SCN to achieve the desired level of performance (e.g., $\text{TLD} \leq \varrho$, $U \leq \tilde{\varrho}$).

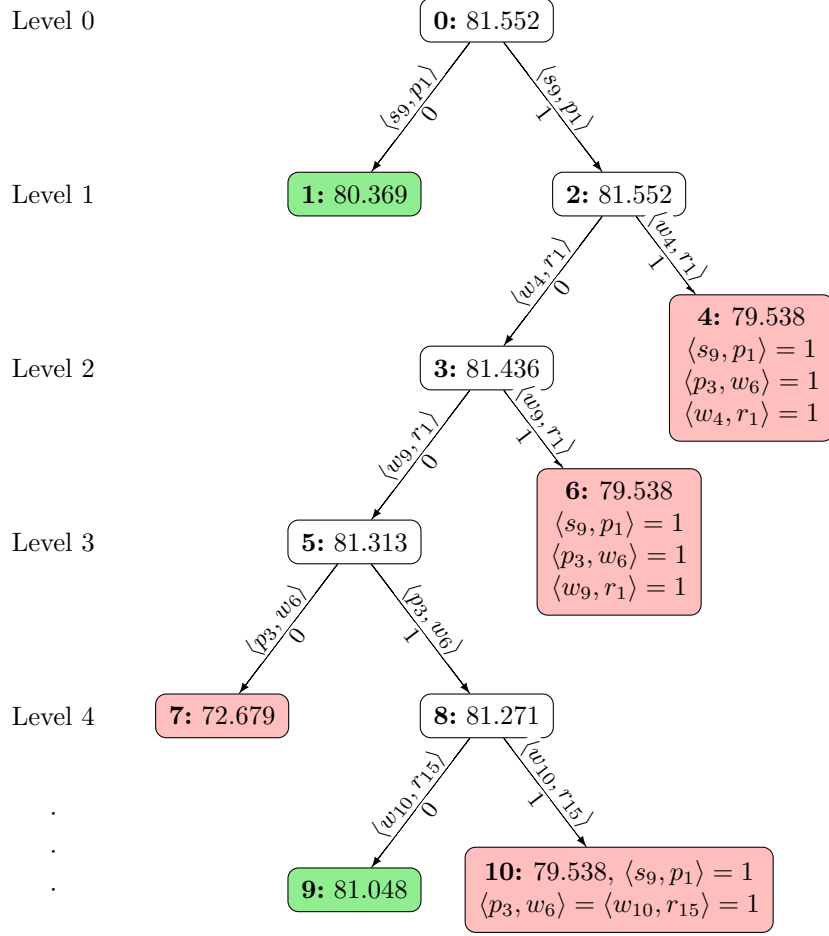


Fig. 3: Algorithm: Identifying Bad-link failure scenarios

5.3 Worst-case analysis of network performance across uncertain demands

In this section, we evaluate the effectiveness of our approach in validating SCNs for metrics **B1** (in Table 7) and **B2** (in Table 8) under uncertain retailer demand requirements as described by Case **A2**. Using the definition of the set $\mathcal{X}$ in (17), we create four testing instances for the demand uncertainty set $\mathcal{X}$ by varying T (the set of past time instances) and $\{\delta^t\}_{t \in T}$ (the observed demands at those time instances) as follows:

1. Chose T such that $|T| = 30$, $\delta^t \sim \text{Uniform}(20, 50)$ for all $t \in T$,
2. Chose T such that $|T| = 40$, $\delta^t \sim \text{Uniform}(40, 90)$ for all $t \in T$,
3. Chose T such that $|T| = 50$, $\delta^t \sim \text{Uniform}(300, 500)$ for all $t \in T$,
4. Chose T such that $|T| = 100$, $\delta^t \sim \text{Uniform}(350, 550)$ for all $t \in T$.

In Columns labeled “T(D)” in Tables 7 and 8, T - denotes the SCN topology among $\{T_1, \dots, T_7\}$ under consideration, whereas (D) - denotes the testing instance among $\{1., 2., 3., 4.\}$ used to construct the demand uncertainty set $\mathcal{X}$ (as described above). In Table 7, Column labeled “NLP”, we report the worst case utilization of the most congested link in the network (U), across all the demand scenarios defined by $\mathcal{X}$, by solving Slack- $\mathcal{M}_d$, which is the single stage formulation for $\max_{x^d \in \mathcal{X}} \text{Slack-M}_d(x^d)$. We describe the construction of Slack- $\mathcal{M}_d$ in Section 4.3. Further, in the Column labeled “RLT”, we report the optimal values for U obtained by solving the first-level RLT relaxation of Slack- $\mathcal{M}_d$ constructed using the Steps F1-F2. In Table 8, the Column labeled “NLP”, reports the optimal values for the worst case total lost demand (TLD), across all the demand scenarios defined by the set $\mathcal{X}$, by solving Slack- $\mathcal{L}_d$, which is the single stage formulation for $\max_{x^d \in \mathcal{X}} \text{Slack-L}_d(x^d)$. We describe the construction of Slack- $\mathcal{L}_d$ in Section 4.3. Further, in the Column labeled “RLT”, we report the optimal values of TLD obtained by solving the first-level RLT relaxation of Slack- $\mathcal{L}_d$ constructed using the Steps F1-F2.

Computational time. From our experiments in Tables 7 and 8, we observe that the computational time stays stable for solving the first-level RLT relaxation, but may explode for the NLP. Since many NLP instances, did not finish even after several hours, we set a 10-hour limit for the solver and reported the lower (LB) - upper (UB) bound for the optimal objective value at termination. We also report the optimality gap, computed as $\text{gap} = \frac{\text{UB} - \text{LB}}{\text{UB}} \times 100$ for each of these NLP instances, that did not converge within 10 hours of runtime (see e.g., $T_3(4.)$, $T_4(4.)$, $T_5(3. - 4.)$, $T_6(1. - 4.)$, and $T_7(1. - 4.)$ in Table 7 and $T_7(1.)$ in Table 8).

Computational accuracy. Consistent with Proposition 4, the testing instances for which the NLP converged to optimality, confirm that our proposed first-level RLT relaxation achieves the optimal value of $\max_{x^d \in \mathcal{X}} \text{Slack-M}_d(x^d)$ in Table 7 and $\max_{x^d \in \mathcal{X}} \text{Slack-L}_d(x^d)$ in Table 8. Moreover, similar to Tables 5 and 6, the results in Tables 7 and 8, can be mutually validated to verify their accuracy and consistency. For e.g., the testing instances in Table 7, with $U^* \leq 1$ and $U^* > 1$, correspond to $\text{TLD}^* = 0$ and $\text{TLD}^* > 0$ respectively in Table 8. This is true since in Table 8, the optimal values of TLD are computed ensuring U is almost one (see the capacity constraint in (8)).

5.4 Capacity augmentation to handle uncertain failures

Our robust certification framework guides SCN designers in how best to augment link capacities to guarantee a given level of SCN performance (e.g., $\text{TLD} \leq \varrho$, $U \leq \tilde{\varrho}$). As discussed in Section 4.5, for $\mathcal{X}$ as defined in (13), the RLT relaxation of $\max_{x^f \in \mathcal{X}} \mathcal{A}_D(x^f)$ allows us to formulate the link capacity augmentation problem as a single LP that always guarantee a conservative robust SCN design quickly. To determine the optimal values for the variable incremental link capacities $\{\epsilon_{ij}\}_{\langle i,j \rangle \in E}$ defined in (19) and (20), we note that for a given $x^f \in \mathcal{X}$, $\{\epsilon_{ij}\}_{\langle i,j \rangle \in E}$ are the dual variables for the constraints in (21b). Leveraging this fact at the optimality of the first-level RLT relaxation of $\max_{x^f \in \mathcal{X}} \mathcal{A}_D(x^f)$, we determine the incremental capacity for each link $\langle i, j \rangle$ as the dual variable for the Constraint $\alpha_{ij}U' - U'\alpha_{ij}^x \leq \rho_{ij}$ (21b), where α_{ij}^x linearizes $\alpha_{ij}x_{ij}$ for all $\langle i, j \rangle \in E$. Computationally, we determine these

dual values, ϵ_{ij} , using the Gurobi API function `GRBModel::getConstrByName()` and the Gurobi attribute `.Pi` (see [52] for reference).

In Table 9, we report the results for our link capacity augmentation scheme that ensures $U^* \leq 1$, amidst link failure scenarios in (13). In the Column labeled “T (f_1, f_2, f_3)”, we report the SCN and the link failure scenarios under consideration. The Column labeled “ U_B ” reports the optimal value of U obtained via the first-level RLT relaxation of Slack- $\mathcal{M}_f$ with the original link capacities $\{c_{ij}\}_{\langle i,j \rangle \in E}$ (see Table 5). In Column “Additional-Capacities (ϵ)”, we report the values for $\{\epsilon_{ij}\}_{\langle i,j \rangle \in E}$ obtained as described in the above paragraph. Then, for all $\langle i, j \rangle \in E$, the updated link capacity c'_{ij} is obtained as $c'_{ij} = c_{ij} + \epsilon_{ij}$. Finally in the Column labeled “ U_A ”, we provide the upper bound on U obtained by the first-level RLT relaxation of Slack- $\mathcal{M}_f$ with the updated link capacities $\{c'_{ij}\}_{\langle i,j \rangle \in E}$. A value of 0.9999 in Column “ U_A ”, shows that the described capacity augmentation scheme indeed results in $U \leq 1$.

6 Conclusion

In this paper, we have made significant strides in the field of supply chain network (SCN) design by introducing a robust optimization framework. This framework allows SCN designers to certify whether a given design performs acceptably for chosen performance measures across a set of uncertain network link failures and demand requirements, which may be exponential and non-enumerable. Our approach has demonstrated its practical applicability; the first-level RLT relaxation can provably solve the robust certification problem for predicted demands and consistently provides reliable overestimates for the cases involving failures. The empirical results demonstrate the computational stability of our techniques and also highlight the infeasibility of solving the certification problems using traditional ways. The insights gained from our framework enable designers to understand SCN performance under uncertain network disruptions, guide incremental design refinements, and identify performance-critical sections of the SCN, thus contributing to the development of SCNs that are robust to failures and variable demands. Furthermore, we have proposed a branch-and-bound (B-B) algorithm that aids in identifying link failure scenarios that result in poor SCN performance. Identifying such scenarios in advance is crucial for enhancing the robustness of the SCN. Combined with our capacity augmentation scheme, this B-B algorithm helps improve the overall performance of the SCN in a cost-efficient manner. Encouraged by our initial results, we plan to explore larger supply chains and consider SCN design more extensively in the future to build more reliable networks. Our work lays a solid foundation for future research and practical applications in reliability evaluation and supply chain resilience.

References

- [1] Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E.: Designing and Managing the Supply Chain: Concepts, Strategies, and Cases. McGraw-hill, New York, USA (1999)

- [2] Shukla, A., Agarwal Lalit, V., Venkatasubramanian, V.: Optimizing efficiency-robustness trade-offs in supply chain design under uncertainty due to disruptions. *International Journal of Physical Distribution & Logistics Management* **41**(6), 623–647 (2011)
- [3] Zhao, K., Scheibe, K., Blackhurst, J., Kumar, A.: Supply chain network robustness against disruptions: Topological analysis, measurement, and optimization. *IEEE Transactions on Engineering Management* **66**(1), 127–139 (2018)
- [4] Nagurney, A.: Optimal supply chain network design and redesign at minimal total cost and with demand satisfaction. *International Journal of Production Economics* **128**(1), 200–208 (2010)
- [5] Ivanov, D., Dolgui, A., Sokolov, B., Ivanova, M.: Literature review on disruption recovery in the supply chain. *International Journal of Production Research* **55**(20), 6158–6174 (2017)
- [6] Katsaliaki, K., Galetsi, P., Kumar, S.: Supply chain disruptions and resilience: A major review and future research agenda. *Annals of Operations Research*, 1–38 (2022)
- [7] Suryawanshi, P., Dutta, P.: Optimization models for supply chains under risk, uncertainty, and resilience: A state-of-the-art review and future research directions. *Transportation research part e: logistics and transportation review* **157**, 102553 (2022)
- [8] Govindan, K., Fattahi, M., Keyvanshokoo, E.: Supply chain network design under uncertainty: A comprehensive review and future research directions. *European journal of operational research* **263**(1), 108–141 (2017)
- [9] Gereffi, G., Lee, J.: Why the world suddenly cares about global supply chains. *Journal of supply chain management* **48**(3), 24–32 (2012)
- [10] Queiroz, M.M., Ivanov, D., Dolgui, A., Fosso Wamba, S.: Impacts of epidemic outbreaks on supply chains: mapping a research agenda amid the covid-19 pandemic through a structured literature review. *Annals of operations research* **319**(1), 1159–1196 (2022)
- [11] Pellegrino, R., Gaudenzi, B.: Impacts and supply chain resilience strategies to cope with covid-19 pandemic: a literature review. *Supply Chain Resilience*, 5–18 (2023)
- [12] Chen, H.Y., Das, A., Ivanov, D.: Building resilience and managing post-disruption supply chain recovery: Lessons from the information and communication technology industry. *International Journal of Information Management* **49**, 330–342 (2019)

- [13] Fahimnia, B., Jabbarzadeh, A., Sarkis, J.: Greening versus resilience: A supply chain design perspective. *Transportation Research Part E: Logistics and Transportation Review* **119**, 129–148 (2018)
- [14] Bertsimas, D., Thiele, A.: A robust optimization approach to inventory theory. *Operations research* **54**(1), 150–168 (2006)
- [15] Tang, C.S.: Robust strategies for mitigating supply chain disruptions. *International Journal of Logistics: Research and Applications* **9**(1), 33–45 (2006)
- [16] You, F., Grossmann, I.E.: Design of responsive supply chains under demand uncertainty. *Computers & Chemical Engineering* **32**(12), 3090–3111 (2008)
- [17] Peng, P., Snyder, L.V., Lim, A., Liu, Z.: Reliable logistics networks design with facility disruptions. *Transportation Research Part B: Methodological* **45**(8), 1190–1211 (2011)
- [18] Qiu, R., Wang, Y.: Supply chain network design under demand uncertainty and supply disruptions: a distributionally robust optimization approach. *Scientific Programming* **2016**(1), 3848520 (2016)
- [19] Yeh, C.-T., Lyu, S.-H., Fiondella, L.: Power supply reliability assessment for a multistate electrical power network with line loss rates. *Annals of Operations Research*, 1–22 (2024)
- [20] Chang, Y., Jiang, C., Chandra, A., Rao, S., Tawarmalani, M.: Lancet: Better network resilience by designing for pruned failure sets. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* **3**(3), 1–26 (2019)
- [21] Chandra, A., Tawarmalani, M.: Probability estimation via policy restrictions, convexification, and approximate sampling. *Mathematical Programming* **196**(1), 309–345 (2022)
- [22] Shishebori, D., Babadi, A.Y.: Robust and reliable medical services network design under uncertain environment and system disruptions. *Transportation Research Part E: Logistics and Transportation Review* **77**, 268–288 (2015)
- [23] Chang, P.-C., Huang, D.-H., Huang, C.-F.: Simulation-based system reliability estimation of a multi-state flow network for all possible demand levels. *Annals of Operations Research*, 1–16 (2024)
- [24] Huang, C.-F., Huang, D.-H., Lin, Y.-K., Chen, Y.-F.: Network reliability evaluation of manufacturing systems by using a deep learning approach. *Annals of Operations Research*, 1–18 (2022)
- [25] Bertsimas, D., Brown, D.B., Caramanis, C.: Theory and applications of robust optimization. *SIAM review* **53**(3), 464–501 (2011)

- [26] Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust optimization (2009)
- [27] Lee, J., Leyffer, S.: Mixed Integer Nonlinear Programming vol. 154. Springer, USA (2011)
- [28] Wang, H., Xie, H., Qiu, L., Yang, Y.R., Zhang, Y., Greenberg, A.: Cope: Traffic engineering in dynamic networks. In: Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp. 99–110 (2006)
- [29] Adenso-Díaz, B., Mar-Ortiz, J., Lozano, S.: Assessing supply chain robustness to links failure. *International Journal of Production Research* **56**(15), 5104–5117 (2018)
- [30] Ramaekers, K., Janssens, G.K., Sörensen, K., Van Landeghem, R.: A verification method for the simulation of supply chain networks with unreliable links. *Int. Journal of Simulation* **8**(1), 39–47 (2007)
- [31] Baghalian, A., Rezapour, S., Farahani, R.Z.: Robust supply chain network design with service level against disruptions and demand uncertainties: A real-life case. *European journal of operational research* **227**(1), 199–215 (2013)
- [32] Pan, F., Nagi, R.: Robust supply chain design under uncertain demand in agile manufacturing. *Computers & operations research* **37**(4), 668–683 (2010)
- [33] Mo, Y., Harrison, T.P.: A conceptual framework for robust supply chain design under demand uncertainty. In: *Supply Chain Optimization*, pp. 243–263. Springer, USA (2005)
- [34] Khalifehzadeh, S., Seifbarghy, M., Naderi, B.: A four-echelon supply chain network design with shortage: Mathematical modeling and solution methods. *Journal of Manufacturing Systems* **35**, 164–175 (2015)
- [35] Raffei, H., Safaei, F., Rabbani, M.: Integrated production-distribution planning problem in a competition-based four-echelon supply chain. *Computers & Industrial Engineering* **119**, 85–99 (2018)
- [36] Gharaei, A., Pasandideh, S.H.R., Arshadi Khamseh, A.: Inventory model in a four-echelon integrated supply chain: Modeling and optimization. *Journal of Modelling in Management* **12**(4), 739–762 (2017)
- [37] Olivares Vera, D.A., Olivares-Benitez, E., Puente Rivera, E., López-Campos, M., Miranda, P.A.: Combined use of mathematical optimization and design of experiments for the maximization of profit in a four-echelon supply chain. *Complexity* **2018**, 1–25 (2018)
- [38] Badejo, O., Ierapetritou, M.: A mathematical modeling approach for supply chain

- management under disruption and operational uncertainty. *AIChE Journal* **69**(4), 18037 (2023)
- [39] Sebatjane, M., Adetunji, O.: A four-echelon supply chain inventory model for growing items with imperfect quality and errors in quality inspection. *Annals of Operations Research* **335**(1), 327–359 (2024)
 - [40] Tsiakis, P., Shah, N., Pantelides, C.C.: Design of multi-echelon supply chain networks under demand uncertainty. *Industrial & engineering chemistry research* **40**(16), 3585–3604 (2001)
 - [41] Sabri, E.H., Beamon, B.M.: A multi-objective approach to simultaneous strategic and operational planning in supply chain design. *Omega* **28**(5), 581–598 (2000)
 - [42] Bertsimas, D., Goyal, V.: On the power and limitations of affine policies in two-stage adaptive optimization. *Mathematical programming* **134**(2), 491–531 (2012)
 - [43] Bertsimas, D., Sim, M.: The price of robustness. *Operations research* **52**(1), 35–53 (2004)
 - [44] Ben-Tal, A., Nemirovski, A.: Robust solutions of uncertain linear programs. *Operations research letters* **25**(1), 1–13 (1999)
 - [45] DANTZIG, G.B.: *Linear Programming and Extensions*. Princeton University Press, USA (1991)
 - [46] Bradley, S.P., Hax, A.C., Magnanti, T.L.: *Applied Mathematical Programming*. Addison-Wesley Pub. Co., USA (1977)
 - [47] Serali, H.D., Adams, W.P.: *A Reformulation-linearization Technique for Solving Discrete and Continuous Nonconvex Problems* vol. 31. Springer, USA (2013)
 - [48] Handelman, D.: Representing polynomials by positive linear functions on compact convex polyhedra. *Pacific Journal of Mathematics* **132**(1), 35–62 (1988)
 - [49] Nie, T., Guo, Z., Zhao, K., Lu, Z.-M.: New attack strategies for complex networks. *Physica A: Statistical Mechanics and its Applications* **424**, 248–253 (2015)
 - [50] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms*, 3rd Edition. MIT Press, USA (2009)
 - [51] Boyd, S., Vandenberghe, L.: *Convex Optimization*. Cambridge University Press, USA (2004)
 - [52] Gurobi Optimization, LLC: *Gurobi Optimizer Reference Manual* (2023). <https://www.gurobi.com>

T(D)	NLP	T-NLP	RLT	T-RLT	T-Dem%	V-Dev%
T ₁ (1.)	1.470	0.04	1.470	0.04	0	0
T ₁ (2.)	2.674	0.04	2.674	0.03	25	0
T ₁ (3.)	14.888	0.06	14.888	0.06	0	0
T ₁ (4.)	16.384	0.48	16.384	0.11	77.08	0
T ₂ (1.)	0.324	0.12	0.324	0.08	33.33	0
T ₂ (2.)	0.583	0.21	0.583	0.10	52.38	0
T ₂ (3.)	3.245	0.25	3.245	0.12	52	0
T ₂ (4.)	3.576	0.53	3.576	0.13	75.47	0
T ₃ (1.)	0.858	11.84	0.858	0.32	97.28	0
T ₃ (2.)	1.564	8.96	1.564	0.44	95.09	0
T ₃ (3.)	9.074	9.76	9.074	0.55	94.36	0
T ₃ (4.)	(10.066, 10.292, 2.20%)	36000	10.080	1.05	-	-
T ₄ (1.)	1.995	91.58	1.995	0.23	99.75	0
T ₄ (2.)	3.876	56.46	3.876	0.33	99.42	0
T ₄ (3.)	22.603	5953.21	22.603	0.39	99.99	0
T ₄ (4.)	(25.138, 26.213, 4.10%)	36000	25.138	0.80	-	-
T ₅ (1.)	0.236	11.32	0.236	2.99	73.59	0
T ₅ (2.)	0.427	14.15	0.427	4.90	65.37	0
T ₅ (3.)	(2.438, 3.378, 27.83%)	36000	2.438	4.80	-	-
T ₅ (4.)	(2.499, 3.317, 24.66%)	36000	2.683	11.04	-	-
T ₆ (1.)	(1.267, 1.451, 12.69%)	36000	1.267	1.73	-	-
T ₆ (2.)	(2.327, 2.667, 12.75%)	36000	2.327	2.08	-	-
T ₆ (3.)	(13.825, 15.228, 9.21%)	36000	13.826	2.91	-	-
T ₆ (4.)	(15.715, 17.051, 7.84%)	36000	15.715	6.03	-	-
T ₇ (1.)	(0.851, 1.045, 18.56%)	36000	0.907	1.03	-	-
T ₇ (2.)	(1.565, 1.895, 17.41%)	36000	1.636	1.32	-	-
T ₇ (3.)	(9.468, 10.675, 11.31%)	36000	9.468	1.63	-	-
T ₇ (4.)	(10.789, 11.891, 9.27%)	36000	10.789	5.91	-	-

Table 7: Solving for $U^* = \text{Slack-}\mathcal{M}_d = \max_{x^d \in \mathcal{X}} \text{Slack-M}_d(x^d)$, $\mathcal{X}$ as in (17)

¹ T-NLP: CPU run-time (seconds) for getting the value of U reported in Column “NLP”.

² T-RLT: CPU run-time (seconds) for getting the value of U reported in Column “RLT”.

³ T-Dev% = $\frac{\text{T-NLP} - \text{T-RLT}}{\text{T-NLP}} \times 100$, V-Dev% = $\frac{\text{RLT} - \text{NLP}}{\text{RLT}} \times 100$.

⁴ The tuple entries in Column “NLP”: (LB, UB, gap).

T(D)	NLP	T-NLP	RLT	T-RLT	T-Dev%	V-Dev%
T ₁ (1.)	15.689	0.20	15.689	0.11	45	0
T ₁ (2.)	81.009	0.21	81.009	0.13	38.10	0
T ₁ (3.)	876.158	0.21	876.158	0.14	33.33	0
T ₁ (4.)	976.158	0.45	976.158	0.31	31.11	0
T ₂ (1.)	0	0.74	0	0.42	43.24	0
T ₂ (2.)	0	0.84	0	0.63	25	0
T ₂ (3.)	1227.662	0.90	1227.662	0.71	21.11	0
T ₂ (4.)	1427.662	1.54	1427.662	1.30	15.58	0
T ₃ (1.)	0	0.59	0	0.33	44.07	0
T ₃ (2.)	196.281	0.95	196.281	0.43	54.74	0
T ₃ (3.)	2808.832	1.02	2808.832	0.47	53.92	0
T ₃ (4.)	3158.832	2.51	3158.832	0.90	64.14	0
T ₄ (1.)	176.621	1.03	176.621	0.22	78.64	0
T ₄ (2.)	510.617	1.25	510.617	0.34	72.80	0
T ₄ (3.)	3835.048	1.45	3835.048	0.39	73.10	0
T ₄ (4.)	4285.048	2.75	4285.048	0.68	75.27	0
T ₅ (1.)	0	4.89	0	2.49	49.08	0
T ₅ (2.)	0	4.54	0	3.16	30.40	0
T ₅ (3.)	1468.591	9.12	1468.591	6.48	28.95	0
T ₅ (4.)	1902.422	29.03	1902.422	19.22	33.79	0
T ₆ (1.)	124.178	4.56	124.178	1.54	66.23	0
T ₆ (2.)	616.869	6.09	616.868	1.92	68.47	0
T ₆ (3.)	5962.045	12.30	5962.045	7.70	37.40	0
T ₆ (4.)	6840.593	10.71	6840.593	4.47	58.26	0
T ₇ (1.)	(0, 1.472, 100%)	-	0	2.98	-	-
T ₇ (2.)	433.561	18.64	433.561	1.69	90.93	0
T ₇ (3.)	6494.620	21.26	6494.620	2.12	90.03	0
T ₇ (4.)	7507.526	30.71	7507.526	5.85	80.95	0

Table 8: Solving for TLD* = Slack- $\mathcal{L}_d$ ensuring $U \leq 1$, $\mathcal{X}$ in (17)

¹ Slack- $\mathcal{L}_d = \max_{x^d \in \mathcal{X}} \text{Slack-L}_d(x^d)$.

² T-NLP: Run-time (seconds) for obtaining the value of TLD in Column “NLP”.

³ T-RLT: Run-time (seconds) for obtaining the value of TLD in Column “RLT”.

⁴ T-Dev% = $\frac{\text{T-NLP} - \text{T-RLT}}{\text{T-NLP}} \times 100$, V-Dev% = $\frac{\text{RLT} - \text{NLP}}{\text{RLT}} \times 100$.

⁵ The tuple entries in Column “NLP”: (LB, UB, gap).

T (f_1, f_2, f_3)	U_B	Additional-Capacity (ϵ)	U_A
T ₁ (2, 2, 1)	2.084	$\epsilon_{\langle w_1, r_2 \rangle} = 12.83, \epsilon_{\langle w_2, r_1 \rangle} = 2.48$ $\epsilon_{\langle w_2, r_2 \rangle} = 3.10$	0.9999
T ₃ (1, 2, 2)	1.682	$\epsilon_{\langle p_2, w_3 \rangle} = 3.96, \epsilon_{\langle p_2, w_4 \rangle} = 5.71$ $\epsilon_{\langle p_3, w_1 \rangle} = 3.21, \epsilon_{\langle w_1, r_1 \rangle} = 5.45$ $\epsilon_{\langle w_1, r_7 \rangle} = 9.53, \epsilon_{\langle w_2, r_6 \rangle} = 6.53$ $\epsilon_{\langle w_3, r_6 \rangle} = 3.85$	0.9999
T ₆ (1, 1, 1)	1.217	$\epsilon_{\langle s_3, p_1 \rangle} = 10.84, \epsilon_{\langle s_7, p_1 \rangle} = 15.90$ $\epsilon_{\langle p_3, w_{10} \rangle} = 6.97, \epsilon_{\langle p_2, w_1 \rangle} = 19.78$ $\epsilon_{\langle p_2, w_{10} \rangle} = 6.09, \epsilon_{\langle p_3, w_1 \rangle} = 18.14$ $\epsilon_{\langle p_3, w_5 \rangle} = 0.09, \epsilon_{\langle p_3, w_8 \rangle} = 4.47$	0.9999
T ₇ (0, 2, 1)	3.366	$\epsilon_{\langle p_2, w_2 \rangle} = 16.62, \epsilon_{\langle p_3, w_3 \rangle} = 4.85$ $\epsilon_{\langle p_4, w_5 \rangle} = 5.63, \epsilon_{\langle p_5, w_1 \rangle} = 39.23$ $\epsilon_{\langle p_6, w_5 \rangle} = 29.55, \epsilon_{\langle p_6, w_8 \rangle} = 5.03$ $\epsilon_{\langle w_1, r_9 \rangle} = 14.79, \epsilon_{\langle w_2, r_2 \rangle} = 11.90$ $\epsilon_{\langle w_2, r_{14} \rangle} = 7.94, \epsilon_{\langle w_3, r_8 \rangle} = 1.68$ $\epsilon_{\langle w_4, r_3 \rangle} = 6.08, \epsilon_{\langle w_5, r_8 \rangle} = 9.32$ $\epsilon_{\langle w_7, r_2 \rangle} = 12.66, \epsilon_{\langle w_7, r_8 \rangle} = 14.42$ $\epsilon_{\langle w_8, r_3 \rangle} = 11.42$	0.9999

Table 9: Capacity augmentation ensuring $U \leq 1$ for failures in $\mathcal{X}$ (13)