Optimization Models for Differentiating Quality of Service Levels in Probabilistic Network Capacity Design Problems

Siqian Shen

Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, MI 48109, siqian@umich.edu

Zhihao Chen

Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, MI 48109, czhihao@umich.edu

Abstract: This paper develops various chance-constrained models for optimizing the probabilistic network design problem (PNDP), where we differentiate the quality of service (QoS) and measure the related network performance under uncertain demand. The upper level problem of PNDP designs continuous/discrete link capacities shared by multi-commodity flows, and the lower level problem differentiates the corresponding QoS for demand satisfaction, to prioritize customers and/or commodities. We consider PNDP variants that have either fixed flows (formulated at the upper level) or recourse flows (at the lower level) according to different applications. We transform each probabilistic model into a mixed-integer program, and derive polynomial-time algorithms for special cases with single-row chance constraints. The paper formulates benchmark stochastic programming models by either enforcing to meet all demand or penalizing unmet demand via a linear penalty function. We compare different models and approaches by testing randomly generated network instances and an instance built on the Sioux-Falls network. Numerical results demonstrate the computational efficacy of the solution approaches and derive managerial insights.

Keywords: Network design problem; Quality of Service (QoS); Chance-constrained programming; Stochastic programming; Mixed-integer linear programming; Multi-commodity network flows

---

1. Introduction

Network design problems (NDPs) are essential and fundamental for the development of modern societies. Related crucial decisions include (i) an allocation of limited resources to expand existing link capacities and/or to add new links, (ii) an assignment of origin–destination (OD) pairs, and (iii) a plan of static/dynamic network flows. The decisions are often made under uncertainty due to demand variation and random network disruptions. Therefore, the objective of stochastic NDPs takes into account probabilistic measures of specific network performance indexes, such as the expected generalized cost, travel time, demand satisfaction rate, etc.

Depending on the nature of capacity design variables, NDPs in general fall into three categories with continuous, discrete (usually 0–1 binary), and mixed decision variables at the design phase, or equivalently speaking, at the *upper* level. For stochastic NDPs, the capacity design decisions are made before knowing the uncertainty, and additional recourse decisions are determined according to specific realizations of uncertain parameter, at the *lower* level. For specific applications, flow decisions between OD pairs may need to be determined *a priori* to the realization of the uncertainty, or could be adjustable so as to maximize the expected profit associated with some network performance index. Respectively, they are either decisions made at the upper level or recourse variables made at the lower level.

1.1 Previous work on network design problems

The NDP has been intensively studied for transportation networks, of which a general formulation determines the expansion or addition of links at the upper level, subject to user equilibrium constraints that reflect the route choice behavior of travelers at the lower level. Wang et al. (2013), Li et al. (2012), Gao et al. (2005), Luathep et al. (2011) apply global optimization methods and meta-heuristic approaches to study discrete, continuous, and mixed NDPs. Stochastic programming approaches (Patil and Ukkusuri 2007, Ukkusuri and Waller 2008) and robust optimization methods (Ukkusuri et al. 2006, Chung et al. 2011, Lou et al. 2009, Yin et al. 2009) have been developed for optimizing dynamic NDPs for unknown demand. The latter often yields relatively conservative solutions for bounding the worse-case objective value over some uncertainty set of random parameters. Sharma et al. (2010) use sampling-based approximation algorithms for computing large-scale non-convex NDP problems. Ukkusuri and Patil (2009) consider multi-time-period network investment decisions as compared to a single-stage capacity expansion, and formulate the corresponding NDP variant as a bi-level stochastic program with complementarity constraints. Overall, Chen et al. (2011) provide a comprehensive review of formulations and methodologies for solving transport NDPs under uncertainty.

The NDP arises in a broader class of network-related applications, including supply chain management (MirHassani et al. 2000, Santoso et al. 2005), emergency response (Chang et al. 2007, Oh and Haghani 1997, Sheu 2007), and shelter relief (Kulshrestha et al. 2011), where the upper-level problem determines continuous or discrete capacity levels, followed by the lower-level problem having flow recourses. The research often formulates two-stage stochastic optimization models and employs sampling techniques, approximation algorithms and/or cutting-plane methods, such as the Benders cuts (see, e.g., MirHassani et al. 2000, Barton et al. 1989), for iteratively optimizing the design and flow solutions.

In general, the NDP has various types, including transport NDPs involving traffic equilibria (if exist) and other NDPs only concerning flow balances between OD pairs. The former usually involve a traffic assignment problem, where individual users' flow decisions are taken into account and use equilibrium constraints are formulated in the mathematical models involved, as each user works to reduce his or her own transport costs given a designed network. Patriksson (1994) discusses various traffic assignment models and provides comprehensive algorithms for computing equilibrium solutions. However, when there is insufficient information on the travel costs faced by individual users, it becomes difficult to determine the equilibrium flow. In some other circumstances, network flows (e.g., shipments from suppliers to customers in a supply chain network) are assigned by a central operator who also design the network and no individual users are involved. The NDPs in these cases only impose balances of single- or multi-commodity flows, but are usually coupled with randomness in the demand data. We consider the latter type of NDPs, in which we design network arcs and balance supply-to-demand flows without user equilibrium constraints.

1.2 Contributions and structure of paper

In this paper, we study probabilistic network design problems (PNDPs) under demand uncertainty and with multi-commodity flows, which can be interpreted as shipments of multiple products, or other types of heterogeneous flows involved in a wide class of applications. The flow variables obey balance constraints and knapsack constraints that limit the summation of all flows within a shared capacity on every arc (e.g., Ahuja et al. 1993). Instead of the expectation-based stochastic programming approach, we use probabilistic constraints, or chance constraints (Charnes et al. 1958) to differentiate demand satisfaction rates of shipping multiple commodities to different locations (nodes). The paper formulates four types of chance-constrained models to allow the flexibility in differentiating Quality of Service (QoS) levels with respect to commodity and node-wise demand.

Following the order of PNDP variants discussed in this paper, we determine both flow variables and continuous capacity expansion variables of existing links at the upper level, and evaluate demand satisfaction rates at the lower level. The aim is to minimize the total cost of capacity expansion and flow assignment subject to various forms of chance constraints for bounding the demand losses. We justify this approach by the application of supply chain design, where a flow scheduler needs to be decided before knowing the uncertainty and cannot be easily adjusted in different scenarios. For instance, due to high contracting fees and the ease of maintaining a relationship with a stable set of suppliers and customers, the scheduler may prefer a fixed shipping schedule regardless of actual daily demand when the demand fluctuation is not significant. In some emergency response applications, reaction time for changing the delivery plan may be too short so that fixed flows are more favorable.

We formulate benchmark stochastic network design problems (SNDPs) by letting flows be recourse variables, whose values are determined *after* knowing the demand. The modification results in the flexibility of having different flow decisions in each scenario, but enforces a hard constraint on the flow decisions. We consider two types of SNDPs, of which one that does not penalize unmet demand, and the other adds linear penalty cost for each unit of unmet demand. The latter is typically used in the existing literature and will provide benchmarks in our numerical results. Furthermore, we consider PNDPs and SNDPs with discrete (binary) design variables, representing the addition of new links, and reformulate all formulation variants under this assumption.

Our paper adopts a static modeling framework, and studies the NDP with multi-commodity flows without user equilibrium constraints. We transform all probabilistic models into equivalent mixed-integer programming (MIP) formulations under the assumption of finitely distributed random demand. For some special cases of the PNDP, we present alternative polynomial-time algorithms. The paper demonstrates the relationship of PNDP variants via their risk parameters, e.g., the reliability levels associated with various chance constraints. For large-scale PNDP/SNDP models, we describe a general Benders decomposition approach to improve the computational efficiency. We derive managerial insights by computing on randomly generated network instances and the Sioux-Falls network.

The main contribution of the paper is to formulate PNDPs with various forms of chance constraints to differentiate the reliability and QoS, of which the corresponding research has not been well developed in the literature. To our best knowledge on related work, Lo and Tung (2003) study the tradeoff between the maximum flow in a network and the extent of satisfying chance constraints of the probabilistic user equilibrium, given that link capacities are subject to stochastic degradations. Chen and Yang (2004) account for both spatial equity and demand uncertainty, and formulate the equity constraint as a chance constraint. Waller and Ziliaskopoulos (2001) formulate a chance-constrained model for studying continuous NDPs with traffic dynamics and random time-dependent demands.

1.3 Notation and assumptions

Here we provide a full list of notation and assumptions that are used throughout the paper. We formulate the NDPs on a directed connected graph $G(N, A)$, where N is the set of nodes and $A \subset N \times N$ represents the set of links.

Sets

W	Set of commodities
$O_w \subseteq N$	Set of origins of commodity $w \in W$
$D_w \subseteq N$	Set of destinations of commodity $w \in W$
Ω	Set of random scenarios where $\Omega = \{1, \ldots, \lvert\Omega\rvert\}$

Parameters

c_{ij}	Cost of allocating one unit of capacity at link $(i, j) \in A$
q_{ij}	Fixed cost of adding link $(i, j) \in A$ when capacity design variables are binary
a_{ijw}	Unit cost of flowing commodity $w \in W$ on link $(i, j) \in A$
u_{ij}	Fixed capacity of link $(i, j) \in A$ when capacity design variables are binary
v_{iw}	Unit penalty cost of unmet demand of commodity w at destination $i \in D_w$
o_{iw}	Deterministic supply of commodity w at origin $i \in O_w$
d_{iw}	Random demand of commodity w at destination $i \in D_w$
ξ^s_{iw}	Realization of random demand d_{iw} in scenario $s \in \Omega$, $\forall w \in W$ and $i \in D_w$
p^s	Probability of scenario $s \in \Omega$
$\epsilon, \epsilon_{iw}, \epsilon_i, \epsilon_w$	Risk parameters associated with different forms of chance constraints

Decision variables

$x_{ij} \geq 0$	Continuous capacity of link $(i, j) \in A$ shared by all commodities
$\beta_{ij} \in \{0, 1\}$	Indicating whether to add link $(i, j) \in A$ such that $\beta = 1$ if yes, and 0 otherwise
$y_{ijw} \geq 0$	Continuous fixed flow of commodity $w \in W$ on link $(i, j) \in A$
$y^s_{ijw} \geq 0$	Continuous recourse flow of commodity $w \in W$ on link $(i, j) \in A$ in scenario $s \in S$
$t^s_{iw} \geq 0$	Unmet demand of commodity w at destination $i \in D_w$ in scenario $s \in S$
$z^s \in \{0, 1\}$	Indicating whether the joint chance constraint gets satisfied in scenario $s \in S$, such that $z^s = 1$ if yes, and $z^s = 0$ otherwise

Without loss of generality, we assume that both o_{iw} and d_{jw} are positive integers. The supply vector $\mathbf{o}$ is deterministic, while the demand vector $\mathbf{d}^T = [d_{iw} : w \in W,\ i \in D_w]$ is random, following a known joint discrete distribution among all commodities and their corresponding destinations.

We name all models considered this paper as "`Problem-Capacity-Constraint`," where the types of `Problem` are either "PNDP" (flow fixed at the upper level) or "SNDP" (flow recourse available at the lower level), and the types of `Capacity` are either "-cont" (continuous values of capacity decisions at existing links) or "-bin" (binary decisions of adding new links). The paper refers to "-joint," "-nc," "-c," and "-n" as the types of `Constraint` for PNDPs, representing the cases where the chance constraints are formulated with respect to the overall joint, node-commodity-wise, commodity-wise, and node-wise probabilistic restrictions of unsatisfied demand, respectively; it also refers to "-wop" and "-wp" as the types of `Constraint` for SNDPs, representing the cases where unmet demand is not penalized and penalized in the objective, respectively.

The remainder of the paper is organized as follows. Section 2 assumes continuous network design variables and formulates various PNDPs with fixed flow variables and SNDPs with recourse flow variables. Section 3 describes NDP formulations with discrete network design variables. Section 4 develops solution methodologies, and demonstrates polynomial-time algorithms for special cases of the PNDP with single-row chance constraints. Two sets of numerical examples are given in Section 5: the first consists of randomized instances to test various PNDP formulations and their algorithms; the second simulates a practical setting with the commonly used Sioux Falls network, in which we compare PNDPs with SNDPs. Section 6 concludes the paper and states future research.

2. NDPs with continuous capacity design variables

We begin with the assumption that all upper-level decision variables are continuous, corresponding to the case where we can smoothly increase the capacity of existing links. This section describes the two cases where the multi-commodity flow variables are fixed before (i.e., PNDP-cont) and decided after (i.e., SNDP-cont) realizing the demand uncertainty.

2.1 Fixed flow variables at the upper level

We first analyze PNDP-cont and give its four variations, where the flow decisions are fixed *before* the realization of demands. The variations are determined by the manner in which the uncertain demand constraints are joined by the chance-constraints: by joining all of them (-joint), node-commodity-wise (-nc), commodity-wise (-c), and node-wise (-n). We first focus on the joint case, then present the remaining variants as multiple chance-constrained models.

2.1.1 Joint chance-constraints

This variant involves a joint chance constraint for measuring demand satisfaction of all commodities at all nodes. The aim is to minimize the total cost of capacity design and network flows, while the probability of no demand loss of every commodity at every node is bounded from below by a given reliability level $1-\epsilon$. The joint chance-constraint is given by

$$\mathbb{P}\left(\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq d_{iw},\ \forall i \in D_w,\ w \in W\right) \geq 1-\epsilon, \tag{1}$$

where $\mathbb{P}(\cdot)$ denotes the probability of uncertain event $\cdot$; $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw}$ represents the overall amount of commodity w received at destination i, for all $w \in W$ and $i \in D_w$.

By assuming discretely distributed demand, we transform all chance-constrained models in this paper as equivalent MIP formulations. Define binary variables z^s indicating whether Constraint (1) is violated in scenario s, such that $z^s = 1$ if it is, and $z^s = 0$ otherwise. We have

[PNDP-cont-joint]:

$$\min \quad \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{w\in W}\sum_{(i,j)\in A} a_{ijw}y_{ijw} \tag{2a}$$

$$\text{s.t.} \quad \sum_{w\in W} y_{ijw} \leq x_{ij} \quad \forall (i,j)\in A \tag{2b}$$

$$\sum_{j:(i,j)\in A} y_{ijw} - \sum_{j:(j,i)\in A} y_{jiw} \leq o_{iw} \quad \forall i \in O_w,\ w\in W \tag{2c}$$

$$\sum_{j:(i,j)\in A} y_{ijw} - \sum_{j:(j,i)\in A} y_{jiw} = 0 \quad \forall i \notin O_w \cup D_w,\ w\in W \tag{2d}$$

$$-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} - \xi^s_{iw} + \mathbf{M}z^s \geq 0 \quad \forall i\in D_w,\ w\in W,\ s\in\Omega \tag{2e}$$

$$\sum_{s\in\Omega} p^s z^s \leq \epsilon, \tag{2f}$$

$$\mathbf{x}\geq 0,\ \mathbf{y}\geq 0,\ \mathbf{z}\in\{0,1\}^{|\Omega|}, \tag{2g}$$

where $\mathbf{M} \in \mathbb{Z}_+^{\sum_{w\in W}|D_w|}$ is a vector whose entries are arbitrary large numbers. Constraints (2b) ensure that the flows of all commodities at arc (i,j) do not exceed allocated capacities, for all $(i,j)\in A$; (2c) and (2d) impose commodity-wise flow balances at origins and transshipment nodes, respectively; (2e) enforces $z^s = 1$ for some scenario $s\in\Omega$, if at least one commodity at one node has a positive demand loss in that scenario. For the convenience of our later comparison of different models, we rewrite (2e) as

$$\left[-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} - \xi^s_{iw},\ \forall i\in D_w,\ w\in W\right] + \mathbf{M}z^s \geq 0 \quad \forall s\in\Omega, \tag{3}$$

emphasizing that each row $-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} - \xi^s_{iw},\ w \in W,\ i \in D_w$ shares a common binary variable z^s with a big M coefficient, for all scenarios $s \in \Omega$. Constraint (2f) guarantees that an expected probability of violation is no more than ϵ. By having integer parameters $\mathbf{c}$, $\mathbf{a}$, $\mathbf{o}$, and $\mathbf{d}$, there exists an optimal solution having integer $\mathbf{x}$- and $\mathbf{y}$-values, and (2e) and (2f) together yield a deterministic equivalence of the joint chance constraint (1).

2.1.2 Multiple chance constraints

In the remaining three variants of PNDP-cont, the joint chance constraint (1) is split into multiple smaller-scale chance constraints. In the first variant PNDP-cont-nc, one single-row chance constraint is imposed for each commodity w and one of its destination $i \in D_w$ with an individual risk tolerance ϵ_{iw}, i.e.

$$\mathbb{P}\left(\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq d_{iw}\right) \geq 1 - \epsilon_{iw},\ \forall i \in D_w,\ w \in W. \tag{4}$$

We motivate the study of (4) as follows. First, it allows unsatisfied demand at "less important" destinations for some "less important" commodities with higher risk tolerances, rather than assume homogeneous demand-satisfaction guarantees everywhere. In other words, we allow to differentiate the importance of products and/or customers by *appropriately* choosing ϵ_{iw} for each combination of commodity w and node i. Second, a decision maker can vary values of $\epsilon_{iw},\ \forall w \in W,\ i \in D_w$, and derive corresponding optimal risk-and-cost tradeoffs. Such a value-varying process can be implemented as sensitivity analysis, or by treating all ϵ_{iw} as decision variables within certain preferable ranges. Thus, we can consider $\mathbf{x}$, $\mathbf{y}$, and ϵ as decision variables, and seek solutions that simultaneously trade off risk and cost. According to similar motivations, the other two variants involve

- A joint chance constraint for each commodity with a risk tolerance ϵ_w (PNDP-cont-c):

$$\mathbb{P}\left(\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq d_{iw},\ \forall i \in D_w\right) \geq 1 - \epsilon_w,\ \forall w \in W. \tag{5}$$

- A joint chance constraint for each destination with a risk tolerance ϵ_i (PNDP-cont-n):

$$\mathbb{P}\left(\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq d_{iw},\ \forall w \in W\right) \geq 1 - \epsilon_i,\ \forall i \in D_w. \tag{6}$$

Constraints (5) and (6) differentiate risk perceptions with respect to different commodities and destinations, respectively. The corresponding models involve multiple joint chance constraints. By assuming a discrete joint demand distribution, we reformulate each model variant as an equivalent MIP, with binary variables indicating violation status of the chance constraints in each scenario. Analogously to the construction of PNDP-cont-joint, the reformulation procedures yield

[PNDP-cont-nc]:

$$\begin{aligned}
\min \quad & \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{w\in W}\sum_{(i,j)\in A} a_{ijw}y_{ijw} \\
\text{s.t.} \quad & \text{(2b)–(2d)} \\
& -\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} - \xi^s_{iw} + M_{iw}z^s_{iw} \geq 0 \quad \forall i \in D_w,\ w\in W,\ s\in\Omega && \text{(7a)} \\
& \sum_{s\in\Omega} p^s z^s_{iw} \leq \epsilon_{iw} \quad \forall w\in W,\ i\in D_w && \text{(7b)} \\
& \mathbf{x}\geq 0,\ \mathbf{y}\geq 0,\ \mathbf{z}_{iw}\in\{0,1\}^{|\Omega|},\ \forall w\in W,\ i\in D_w, && \text{(7c)}
\end{aligned}$$

where M_{iw} is an arbitrary large number for each commodity w and node $i \in D_w$, and binary variable z^s_{iw} takes the value 1 if there exists a demand loss of commodity w at destination i, and 0 otherwise, $\forall i \in D_w,\ w\in W$.

Define binary variables:

- z^s_w: $=1$ if there exists a demand loss of commodity w on at least one destination $i\in D_w$, and $=0$ otherwise, for all scenarios $s\in\Omega$.
- z^s_i: $=1$ if there exists at least one commodity demand loss at node i, and $=0$ otherwise, for all scenarios $s\in\Omega$.

We can formulate the other two PNDP-cont models as:

[PNDP-cont-c]:

$$\begin{aligned}
\min \quad & \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{w\in W}\sum_{(i,j)\in A} a_{ijw}y_{ijw} \\
\text{s.t.} \quad & \text{(2b)–(2d)} \\
& \left[-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} - \xi^s_{iw},\ \forall i\in D_w\right] + \mathbf{M}_w z^s_w \geq 0 \quad \forall w\in W,\ s\in\Omega && \text{(8a)} \\
& \sum_{s\in\Omega} p^s z^s_w \leq \epsilon_w \quad \forall w\in W && \text{(8b)} \\
& \mathbf{x}\geq 0,\ \mathbf{y}\geq 0,\ \mathbf{z}_w\in\{0,1\}^{|\Omega|},\ \forall w\in W, && \text{(8c)}
\end{aligned}$$

and,

[PNDP-cont-n]:

$$\begin{aligned}
\min \quad & \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{w\in W}\sum_{(i,j)\in A} a_{ijw}y_{ijw} \\
\text{s.t.} \quad & \text{(2b)–(2d)} \\
& \left[-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} - \xi^s_{iw},\ \forall w\in W\right] + \mathbf{M}_i z^s_i \geq 0 \quad \forall i\in \bigcup_{w\in W} D_w,\ s\in\Omega && \text{(9a)}
\end{aligned}$$

$$\sum_{s\in\Omega} p^s z_i^s \leq \epsilon_i \quad \forall i \in \cup_{w\in W} D_w \tag{9b}$$

$$\mathbf{x} \geq 0,\ \mathbf{y} \geq 0,\ \mathbf{z}_i \in \{0,1\}^{|\Omega|},\ \forall i \in \bigcup_{w\in W} D_w. \tag{9c}$$

Remark 1. PNDP-cont-c and PNDP-cont-n can be viewed as hybrid versions of PNDP-cont-joint and PNDP-cont-nc. PNDP-cont-joint measures unsatisfied demands simultaneously and homogeneously for all commodities and nodes, being more conservative than PNDP-cont-nc if subject to the same magnitude of risk tolerances. A scenario in PNDP-cont-joint is considered as "failed" when there exists one location with demand shortage for one product. PNDP-cont-nc "frees" the feasible region to some extent (we also need to compare specific values of ϵ and ϵ_{iw}, $\forall i$, w to draw a more precise statement), and separates probabilistic measures for satisfying each individual d_{iw}. If a decision maker is only interested in evaluating the chance of positive demand loss for every commodity, or for every demand location, we formulate PNDP-cont-c and PNDP-cont-n, respectively. Both models are more conservative than PNDP-cont-nc, and less conservative than PNDP-cont-joint in general. □

We describe in Section 4 the methodological details for optimizing the four MIP models and algorithms in polynomial time for optimizing special cases of PNDP-cont in different forms.

2.2 Recourse flow variables at the lower level

Here we consider the case where capacity design decisions are still continuous, but flow decisions are made *after* the realization of demands, or in other words, SNDP-cont. We present two variations of SNDP-cont, one in which there is no penalty imposed on unmet demand, and the other in which a penalty is imposed on demand that is unmet in any of the possible scenarios.

2.2.1 Without penalty on unmet demand

We first highlight the differences between the formulations of SNDP-cont and those of its PNDP-cont counterparts. As flow decisions are made after demand realization, each scenario has a flow decision variable y^s_{ijw}, as opposed to a single flow decision variable y_{ijw} across all scenarios in the fixed flow case. Furthermore, since the demands are already known when the flow decision is to be made, the formulation no longer requires the binary variable $\mathbf{z}$ to indicate whether the flow is feasible in a particular scenario – the flow variables $\mathbf{y}^s$ now must satisfy the constraints with the demand realization ξ^s, for each scenario s. In addition, the objective value considers the *expected* cost of the flow decisions, weighted by the probability of each scenario occurring, as there are now multiple flow costs associated with the demand realization in each scenario. We can write this expectation explicitly as the sum of the flow costs in each scenario weighted by the probability of the scenario occurring.

The following shows the full formulation of the SNDP-cont-wop problem, which holds many similarities to the PNDP-cont-nc problem. We note here that the demand constraints are always node-commodity-wise constraints (no joint constraints), unlike in the PNDP-cont case, where demand constraints could be joined in various ways.

[SNDP-cont-wop]:

$$\min \quad \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{s\in\Omega} p^s \left(\sum_{w\in W} \sum_{(i,j)\in A} a_{ijw} y^s_{ijw} \right) \tag{10a}$$

$$\text{s.t.} \quad \sum_{w\in W} y^s_{ijw} \le x_{ij} \quad \forall (i,j)\in A,\ s\in\Omega \tag{10b}$$

$$\sum_{j:(i,j)\in A} y^s_{ijw} - \sum_{j:(j,i)\in A} y^s_{jiw} \le o_{iw} \quad \forall i\in O_w,\ w\in W,\ s\in\Omega \tag{10c}$$

$$\sum_{j:(i,j)\in A} y^s_{ijw} - \sum_{j:(j,i)\in A} y^s_{jiw} = 0 \quad \forall i\notin O_w\cup D_w,\ w\in W,\ s\in\Omega \tag{10d}$$

$$-\sum_{j:(i,j)\in A} y^s_{ijw} + \sum_{j:(j,i)\in A} y^s_{jiw} \ge \xi^s_{iw} \quad \forall i\in D_w,\ w\in W,\ s\in\Omega \tag{10e}$$

$$\mathbf{x}\ge 0,\ \mathbf{y}^s \ge 0,\ \forall s\in\Omega \tag{10f}$$

2.2.2 With penalty on unmet demand

Here we present the formulation of SNDP which penalizes unfulfilled demands. A variable t^s_{iw} is introduced for every $i\in D_w, w\in W, s\in\Omega$, to represent the amount of demand for commodity w at destination i that is not fulfilled in scenario s by the solution. This unmet demand is penalized by a factor of v_{iw}, a parameter for the model which varies according to how strictly a demand constraint should be followed. We illustrate the use of the penalty term $v_{iw}t^s_{iw}$ in the formulation of SNDP-cont-wp below.

[SNDP-cont-wp]:

$$\min \quad \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{s\in\Omega} p^s \sum_{w\in W} \left(\sum_{(i,j)\in A} a_{ijw} y^s_{ijw} + \sum_{i\in D_w} v_{iw} t^s_{iw} \right) \tag{11a}$$

$$\text{s.t.} \quad \text{(10b), (10c), (10d)}$$

$$-\sum_{j:(i,j)\in A} y^s_{ijw} + \sum_{j:(j,i)\in A} y^s_{jiw} + t^s_{iw} \ge \xi^s_{iw} \quad \forall i\in D_w,\ w\in W,\ s\in\Omega \tag{11b}$$

$$\mathbf{x}\ge 0,\ \mathbf{y}^s \ge 0,\ \mathbf{t}^s\ge 0,\ \forall s\in\Omega \tag{11c}$$

We compute the demand loss t^s_{iw} in scenario $s\in\Omega$ of commodity $w\in W$ at destination $i\in D_w$ according to (11b), and penalize positive demand losses at a unit cost v_{iw} in the objective function (11a). The model is typically used to formulate cost-based NDPs. We later use SNDP-cont-wp as a benchmark against which we compare our PNDP-cont reformulations in Section 5.

3. NDPs with binary design variables

Sometimes, decisions need to be made on whether to build a link, instead of on the capacity of existing links in the network. In this section, we consider such problems by focusing on NDPs with binary capacity design variables, denoted by β_{ij} for all $(i,j) \in A$. If $\beta_{ij} = 1$ we add the link (i,j) with a fixed capacity u_{ij}; otherwise (i,j) does not exist in the network. Let q_{ij} be the cost of adding link (i,j) for all $(i,j) \in A$. We modify the previous formulations with fixed flow variables according to this new definition.

3.1 Fixed flow variables at the upper level

We modify the variants of PNDP-cont to reflect the restriction of capacity design variable to binary variables only. Here use PNDP-cont-joint as an example (the other models can be reformulated in the same manner). The aim is to minimize both arc construction cost and flow cost, subject to a certain probability guarantee for satisfying the overall demand. To modify PNDP-cont-joint to PNDP-bin-joint in (2), we replace $\sum_{(i,j)\in A} c_{ij}x_{ij}$ in the objective by $\sum_{(i,j)\in A} q_{ij}\beta_{ij}$, and change (2b) into $\sum_{w\in W} y_{ijw} \leq u_{ij}\beta_{ij},\ \forall (i,j) \in A$. All other constraints are kept the same in all MIP models involving binary capacity design variables.

[PNDP-bin-joint]:

$$\begin{aligned}
\min \quad & \sum_{(i,j)\in A} q_{ij}\beta_{ij} + \sum_{w\in W}\sum_{(i,j)\in A} a_{ijw}y_{ijw} && \text{(12a)}\\
\text{s.t.} \quad & \sum_{w\in W} y_{ijw} \leq u_{ij}\beta_{ij} \quad \forall (i,j) \in A && \text{(12b)}\\
& \text{(2c)–(2g)} && \\
& \mathbf{y} \geq 0,\ \mathbf{z} \in \{0,1\}^{|\Omega|},\ \beta_{ij} \in \{0,1\} \quad \forall (i,j) \in A && \text{(12c)}
\end{aligned}$$

3.2 Recourse flow variables at the lower level

The formulations of SNDP-bin are again very similar to that of their SNDP-cont counterparts. The only differences are that $q_{ij}\beta_{ij}$ replaces $c_{ij}x_{ij}$ in the objective function, and that the recourse flow variables are bounded above by $u_{ij}\beta_{ij}$ instead of x_{ij}. Here we present the formulation of SNDP-bin-wp as an example for comparison with SNDP-cont-wp.

[SNDP-bin-wp]:

$$\begin{aligned}
\min \quad & \sum_{(i,j)\in A} q_{ij}\beta_{ij} + \sum_{s\in\Omega} p^s \left(\sum_{w\in W}\sum_{(i,j)\in A} a_{ijw}y^s_{ijw} + \sum_{i\in D_w} v_{iw}t^s_{iw} \right) && \text{(13a)}\\
\text{s.t.} \quad & \sum_{w\in W} y^s_{ijw} \leq u_{ij}\beta_{ij} \quad \forall (i,j) \in A,\ s \in \Omega && \text{(13b)}\\
& \text{(10c)–(10d)} \ ; \ \text{(11b)} && \\
& \beta_{ij} \in \{0,1\},\ \mathbf{y}^s \geq 0,\ \mathbf{t}^s \geq 0,\ \forall (i,j) \in A,\ s \in \Omega && \text{(13c)}
\end{aligned}$$

3.3 Comparison of PNDP/SNDP-bin with PNDP/SNDP-cont

When comparing the PNDP/SNDP-bin formulations with their associated PNDP/SNDP-cont formulations, it can be seen that their only difference is in the network design decision variables. However, having binary variables makes PNDP/SNDP-bin formulations MIP, which are much harder to solve compared to the linear programs of the PNDP/SNDP-cont.

Binary variables are easy to branch on, allowing us to use a branch-and-bound approach for optimizing PNDP/SNDP-bin. Furthermore, only β contains binary variables, so by using a Benders decomposition approach, we isolate the binary variables in the master problem, and solve the subproblem as a linear program. More algorithmic details of solving PNDP/SNDP-bin formulations are presented in Appendix A.

4. Algorithms for optimizing NDPs

In this section, we focus on optimizing the variants of PNDP and SNDP. We first derive valid inequalities for solving PNDP-cont-joint, and then describe how to solve PNDP-cont-nc as well as special cases of PNDP-cont-c/n in polynomial time. We discuss the relationship between joint chance-constraints and single-row chance-constraints, which can be used to derive valid bounds in the branch-and-bound scheme. Finally, we describe a Benders decomposition algorithm to optimize the benchmark SNDP formulations.

4.1 Valid inequalities for optimizing MIP reformulations of PNDPs

We first propose valid inequalities that can be generated to strengthen PNDP-cont-joint, which can be considered to be a more general case of its -nc/-c/-n variants. The concepts of *partial orders* and *induced cover sets* are given as follows based on (1) in PNDP-cont-joint. We refer the readers to Ruszczyński (2002) for general definitions of the two concepts.

Definition 1. Given any feasible $\mathbf{y}$, a partial order $\preceq$ defined on scenarios $\{1,\ldots,|\Omega|\}$ satisfies

$$a \preceq b \;\Leftrightarrow\; \xi^a_{iw} + \sum_{j:(i,j)\in A} y_{ijw} - \sum_{j:(j,i)\in A} y_{jiw} \le \xi^b_{iw} + \sum_{j:(i,j)\in A} y_{ijw} - \sum_{j:(j,i)\in A} y_{jiw},\ \forall i \in D_w,\ w \in W.$$

Moreover, given that the node-arc incidence matrix is deterministic, and the uncertainty only exists in $\mathbf{d}$, there exists a partial order $\preceq$ defined on scenarios $\{1,\ldots,|\Omega|\}$, satisfying

$$a \preceq b \;\Leftrightarrow\; \xi^a_{iw} \le \xi^b_{iw},\ \forall w \in W,\ i \in D_w.$$

For PNDP-cont-joint, there exists an optimal solution $(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}})$, in which

$$a \preceq b \;\Rightarrow\; \hat{z}^a \le \hat{z}^b \quad \forall a,\ b \in \{1,\ldots,|\Omega|\},\ a \ne b.$$

This indicates that if the chance constraint is violated in scenario a (i.e., $z^a = 1$), then definitely it will not be satisfied in scenario b (i.e., $z^b = 1$). For all scenarios $s = 1,\ldots,|\Omega|$, define set

$$L^s := \{k \in \{1,\ldots,|\Omega|\}\ :\ s \preceq k\},$$

i.e., the collection of scenarios in which the chance constraint will be violated if it is violated in scenario s.

Definition 2. A set $C \subseteq \{1, \ldots, |\Omega|\}$ is called an induced cover if

$$\Pr\left\{\bigcup_{s \in C} L^s\right\} > \epsilon. \tag{14}$$

An induced cover set C is minimal if for *every* $k \in C$,

$$\Pr\left\{\bigcup_{s \in C} L^s \setminus \{k\}\right\} \leq \epsilon. \tag{15}$$

For any induced cover set C, there exists an inequality of the form

$$\sum_{s \in C} z^s \leq |C| - 1, \tag{16}$$

which is valid to PNDP-cont-joint (see, e.g., Nemhauser and Wolsey 1988, Wolsey 1998). To search for induced cover sets of Constraint (1) based on Definition 2, one can apply MIP techniques as well as heuristics. Adding (16) to PNDP-cont-joint will potentially improve computational performance by tightening the formulation. However, when the problem is large scale, i.e., $|\Omega|$ is large, the number of possible induced cover sets can be exponential. Similar to (16), one can derive cover inequalities for both PNDP-cont-c and PNDP-cont-n, by identifying the induced cover sets associated with each joint chance constraint in (5) and (6), respectively.

4.2 Polynomial time algorithms for PNDP-cont-nc/c/n

In this section, we explore polynomial-time algorithms for special cases of PNDPs with the presence of single-row chance constraints.

4.2.1 A polynomial-time algorithm for PNDP-cont-nc

As an alternative to solving the MIP formulation (7), we transform the constraints of PNDP-cont-nc into deterministic constraints without binary variables z_{iw}, resulting in a reformulation that is equivalent to solving multiple commodity-wise minimum-cost flow subproblems with revised arc costs. Note that each chance constraint in (4) only has a single row. According to Definition 1, we analogously define partial orders $\preceq_{iw}$ for every combination of i and w as follows.

Definition 3. For any given $w \in W$ and $i \in D_w$, a partial order $\preceq_{iw}$ defined on scenarios $\{1, \ldots, |\Omega|\}$ satisfies

$$a \preceq_{iw} b \;\Leftrightarrow\; \xi^a_{iw} + \sum_{j:(i,j) \in A} y_{ijw} - \sum_{j:(j,i) \in A} y_{jiw} \leq \xi^b_{iw} + \sum_{j:(i,j) \in A} y_{ijw} - \sum_{j:(j,i) \in A} y_{jiw}, \text{ for any feasible } \mathbf{y}.$$

Similarly, define the set

$$L_{iw}^s := \{k \in \{1, \ldots, |\Omega|\} \ : \ s \preceq_{iw} k\}, \ \forall s = 1, \ldots, |\Omega|, \ \forall w \in W, \ i \in D_w,$$

as the collection of scenarios in which the demand of commodity w at node i cannot be satisfied if there is already a deficit in some scenario $s \in \Omega$. Let $\mathcal{P}(L_{iw}^s) := \sum_{k \in L_{iw}^s} p^k$, indicating the *minimal* violation probability of the $(i, w)^{\text{th}}$ chance constraint, given that it has been violated in scenario s. The goal, for all $w \in W$ and $i \in D_w$, is to identify threshold values of the violation probability, such that it yields the same optimal solution by assigning ϵ_{iw} any values from the same interval between neighboring thresholds.

For two consecutive scenarios $s-1$ and s in a partial order $\preceq_{iw}$, suppose that $\xi_{iw}^s > \xi_{iw}^{s-1}$. Given some flow solution $\mathbf{y}$, suppose that

$$-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} < \xi_{iw}^s \tag{17}$$

$$\text{and} \quad -\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} \geq \xi_{iw}^{s-1}. \tag{18}$$

Then, the probability of $-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} < d_{iw}$ must be $\mathcal{P}(L_{iw}^s)$. The definitions are demonstrated on a small example as follows.

Example 1. We focus on one dimension of the demand vector $\mathbf{d}$, and assume that d_{iw} is a random scalar from set $\Omega = \{1, 4, 6, 7, 10, 11\}$, with probabilities 0.1, 0.1, 0.2, 0.3, 0.2, and 0.1, respectively. According to Definition 3, the current order is already a partial order $\preceq_{iw}$, and we obtain

$$L_{iw}^1 = \{1,2,3,4,5,6\}, \ L_{iw}^2 = \{2,3,4,5,6\}, \ L_{iw}^3 = \{3,4,5,6\}, \ L_{iw}^4 = \{4,5,6\}, \ L_{iw}^5 = \{5,6\}, \ L_{iw}^6 = \{6\}.$$

If $-\sum_{j:(i,j)\in A} y_{ijw} + \sum_{j:(j,i)\in A} y_{jiw} < \xi_{iw}^1 = 1$, then the chance constraint is violated in all scenarios, and $\mathcal{P}(L_{iw}^1) = 1$. Similarly, we have $\mathcal{P}(L_{iw}^2) = 0.9$, $\mathcal{P}(L_{iw}^3) = 0.8$, $\mathcal{P}(L_{iw}^4) = 0.6$, $\mathcal{P}(L_{iw}^5) = 0.3$, $\mathcal{P}(L_{iw}^6) = 0.1$.

For all combinations of w and i, we sort all scenarios, yielding an ordered set $\mathcal{O}(i, w) = \{s_1, \ldots, s_{|\Omega|}\}$ such that $\xi_{iw}^{s_1} \leq \xi_{iw}^{s_2} \leq \cdots \leq \xi_{iw}^{s_{|\Omega|}}$. Note that the order of scenarios $s_1, \ldots, s_{|\Omega|}$ in each $\mathcal{O}(i, w)$ may be different for different i and w. By considering independent chance constraints for individual (i, w)-pairs, for the notation brevity, we represent $\Omega = \{1, \ldots, |\Omega|\}$ as the resembled set of scenarios in $\mathcal{O}(i, w)$, which may contain different specific scenario sequences for different pairs of i and w. Therefore, for all $w \in W$ and $i \in D_w$, a partial order $\preceq_{iw}$ is given by

$$L_{iw}^{|\Omega|} \subseteq \cdots \subseteq L_{iw}^2 \subseteq L_{iw}^1, \text{ and thus } 1 = \mathcal{P}(L_{iw}^1) \geq \mathcal{P}(L_{iw}^2) \geq \cdots \geq \mathcal{P}(L_{iw}^{|\Omega|}) \geq 0. \tag{19}$$

Moreover, we denote $\mathcal{P}(L_{iw}^{|\Omega|+1}) = 0$, combine identical values of ξ_{iw}^s, $\forall s \in \Omega$, and associate each new scenario with a cumulative probability. This pre-processing procedure may decrease the number of scenarios, i.e., $|\Omega|$. We continue to use $\Omega = \{1, \ldots, |\Omega|\}$, but assume that each scenario $s \in \Omega$ has a unique ξ_{iw}^s, $\forall w$, i, and $p^s > 0$. Thus,

$$L_{iw}^{|\Omega|} \subset \cdots \subset L_{iw}^2 \subset L_{iw}^1, \ 1 = \mathcal{P}(L_{iw}^1) > \mathcal{P}(L_{iw}^2) > \cdots > \mathcal{P}(L_{iw}^{|\Omega|}) > \mathcal{P}(L_{iw}^{|\Omega|+1}) = 0.$$

We now establish the equivalence of a single-row chance constraint and a deterministic constraint.

Theorem 1. For any $\epsilon_{iw} \in [0,1)$ such that $\mathcal{P}(L_{iw}^{k'}) > \epsilon_{iw} \geq \mathcal{P}(L_{iw}^{k'+1})$ for some $k' \in \{1, ..., |\Omega|\}$, the $(i,w)^{\text{th}}$ chance constraint deterministic constraint (4) is equivalent to the deterministic constraint:

$$\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{k'}. \tag{20}$$

In particular, when $\epsilon_{iw} = 1$, the chance constraint (4) is relaxed.

Proof. Given any $0 \leq \epsilon_{iw} < 1$, we have $\mathcal{P}(L_{iw}^{k'}) > \epsilon_{iw} \geq \mathcal{P}(L_{iw}^{k'+1})$. By contradiction, we assume that $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} < \xi_{iw}^{k'}$, yielding that the CCP violation probability (which is $\leq \mathcal{P}(L_{iw}^{k'})$ according to the general definition of $\mathcal{P}(L_{iw}^{s})$) is larger than ϵ_{iw}. This is a contradiction, and thus $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{k'}$.

Now suppose that $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{k'}$ for some $k' \in \{1, \ldots, |\Omega|\}$ is equivalent to the chance constraint. Based on the monotonicity assumption of ξ_{iw} in a partial order $\preceq_{iw}$, $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{k}$, $\forall k = k', k'-1, \ldots, 1$. Because $\mathcal{P}(L_{iw}^{s})$ is strictly decreasing over $s = 1, \ldots, |\Omega|$, the probability of violating the chance constraint is strictly less than $\mathcal{P}(L_{iw}^{k'})$. Moreover, the feasible region of $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{k'+1}$ is contained in the feasible region of $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{k'}$ due to the monotonically increasing ξ_{iw}^{s} on scenarios $s = 1, \ldots, |\Omega|$. Hence, to ensure the best possible solution to PNDP-cont-nc, we enforce $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} < \xi_{iw}^{k'+1}$. By definition, the violation probability is $\mathcal{P}(L_{iw}^{k'+1})$, and to satisfy the chance constraint, we require $\mathcal{P}(L_{iw}^{k'+1}) \leq \epsilon_{iw}$, which shows that (20) is an *iff* equivalence of the chance constraint when $\epsilon_{iw} \in [0,1)$.

When $\epsilon_{iw} = 1$, due to the non-negativity of any probability value, the chance constraint is satisfied by any feasible $\mathbf{y}$. This completes the proof. □

According to Theorem 1, there exist finite probability thresholds for transforming each single-row chance constraint in PNDP-cont-nc. Moreover, after sorting scenarios and combining probabilities, for every s in $\preceq_{iw}$, an induced cover set L_{iw}^{s} contains scenarios $s, s+1, \ldots, |\Omega|$, and the corresponding $\mathcal{P}(L_{iw}^{s}) = \sum_{k=s,s+1,\ldots,|\Omega|} p^k$. Then, compare ϵ_{iw} with adjacent values of $\mathcal{P}(L_{iw}^{s})$ and $\mathcal{P}(L_{iw}^{s+1})$, for all $s = 1, \ldots, |\Omega|$, and allocate ϵ_{iw} to an appropriate interval. By applying Theorem 1, we transform the chance constraint into a deterministic constraint. As all chance constraints for every combination of w and i are independently stated, we repeat the procedures for each $w \in W$ and $i \in D_w$, and solve PNDP-cont-nc as a deterministic formulation. Algorithm 1 elaborates on this idea in greater detail.

Complexity analysis. In Algorithm 1, the transformation steps before Step 7 are polynomial in $|\Omega|$. Thus, the algorithmic complexity is mainly determined by

$$\min\Big\{(2a) : (2b)\text{–}(2d), \text{ and } (21)\Big\},$$

which is equivalent of solving W minimum cost flow problems for all commodities, by using $(c_{ij} + a_{ijw})$ as a revised arc cost for every arc (i,j) with respect to commodity w, $\forall w \in W$. The arguments

Algorithm 1 A polynomial-time algorithm for optimizing PNDP-cont-nc.

Sort ξ_{iw}^s in an ascending order, combine identical values, and re-arrange the scenario number, for all $w \in W$ and $i \in D_w$.

Set $\{1, \ldots, |\Omega|\}_{iw}$ as the new partial order of the $(i, w)^{\text{th}}$ Constraint (4). For every $s \in \{1, \ldots, |\Omega|\}_{iw}$, identify L_{iw}^s, given by

$$L_{iw}^s = \{s, s+1, \ldots, |\Omega|\} \text{ and } \mathcal{P}(L_{iw}^s) := \sum_{k=s}^{|\Omega|} p^k.$$

for all $w \in W$ and $i \in D_w$ **do**

  Identify $s' \in \{1, \ldots, |\Omega|\}$ such that $\mathcal{P}(L_{iw}^{s'}) > \epsilon_{iw} \geq \mathcal{P}(L_{iw}^{s'+1})$.

  Replace the $(i, w)^{\text{th}}$ chance constraint of (4) with

$$\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq \xi_{iw}^{s'}. \tag{21}$$

end for

Solve PNDP-cont-nc as $\Big\{\min\,(2\text{a}) : \text{subject to } (2\text{b})\text{–}(2\text{d}), \text{ and } (21)\Big\}$.

return Optimal solution $(\mathbf{x}^*, \mathbf{y}^*)$ and the minimum cost $\mathbf{cx}^* + \mathbf{ay}^*$.

are given as follows. By assumption, let $c_{ij} \geq 0,\ \forall (i,j) \in A$ and $a_{ijw} \geq 0,\ \forall (i,j) \in A,\ w \in W$. Multiply constraints (2b) by c_{ij}, and we have

$$\sum_{w\in W} c_{ij} y_{ijw} \leq c_{ij} x_{ij},\ \forall (i,j) \in A. \tag{22}$$

Now add up the above terms over all arcs $(i, j) \in A$, and also add $\sum_{w\in W} \sum_{(i,j)\in A} a_{ijw} y_{ijw}$ to both sides, yielding

$$\sum_{w\in W} \sum_{(i,j)\in A} (c_{ij} + a_{ijw}) y_{ijw} \leq \sum_{(i,j)\in A} c_{ij} x_{ij} + \sum_{w\in W} \sum_{(i,j)\in A} a_{ijw} y_{ijw} = \text{Objective (2a)}. \tag{23}$$

Due to the minimizing nature of the objective, the original objective is then equivalent to minimizing

$$\sum_{w\in W} \sum_{(i,j)\in A} (c_{ij} + a_{ijw}) y_{ijw},$$

subject to (2c), (2d), and (21), which only involve variables $\mathbf{y}$. PNDP-cont-nc is then equivalent of solving W minimum cost flow problems with revised arc costs, each of which involves commodity-based constraints (2c), (2d), and (21). The overall computational complexity is linear in the complexity of algorithms used for sorting and for solving minimum-cost flow problems.

4.2.2 Algorithm demonstration

We demonstrate our approaches on an example of the PNDP-cont-nc which contains three single-row chance constraints, and is formulated on a network depicted in Figure 1. Three commodities are shipped from nodes 0, 1, and 2, respectively, all to destination 4, with supply capacities given as $o_{01} = o_{12} = o_{23} = 10$. Demand (d_{41}, d_{42}, d_{43}) is jointly realized from a set

$$\Omega = \{(3,1,10),(4,3,9),(5,5,8),(6,7,7),(7,8,6),(8,6,5),(9,4,4),(10,2,3)\}$$

with an equal probability 1/8 for each realization. The costs are $c_{01} = c_{02} = c_{24} = 1$, $c_{34} = c_{13} = c_{32} = 2$, and for commodities $w = 1, 2, 3$, $a_{ijw} = 0.5,\ 0.2,\ 0.3$ respectively on all arcs $(i, j) \in A$. The risk levels for unsatisfied demand for commodities 1, 2, and 3 are mandated as $\epsilon_1 = 0.2$, $\epsilon_2 = 0.4$, and $\epsilon_3 = 0.3$, respectively.

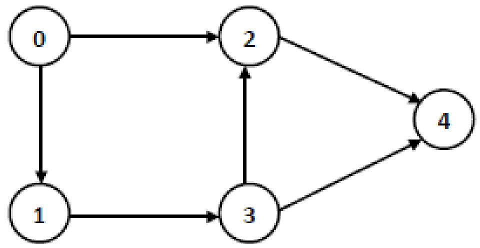


Figure 1: Network topology for the PNDP-cont-nc example

Different from previously presented Algorithm 1, here we directly compute all values of L_w^s based on the definition, without the pre-sorting steps. For commodities 1, 2, and 3,

$$\begin{aligned}
&L_1^1 = \{1,2,\ldots,8\},\ L_1^2 = \{2,\ldots,8\},\ldots,\ L_1^7 = \{7,8\},\ L_1^8 = \{8\},\\
&L_2^1 = \{1,2,\ldots,8\},\ L_2^2 = \{2,\ldots,7\},\ L_2^3 = \{3,\ldots,6\},\ L_2^4 = \{4,5\},\\
&\qquad\qquad L_2^5 = \{5\},\ L_2^6 = \{6,5,4\},\ L_2^7 = \{7,\ldots,3\},\ L_2^8 = \{8,\ldots,2\},\\
&L_3^1 = \{1\},\ L_3^2 = \{1,2\},\ldots,\ L_3^8 = \{1,\ldots,8\}.
\end{aligned}$$

where L_w^s are abbreviations of L_{iw}^s because $i = 4$ is the only destination, $\forall w$. The associated probability thresholds are $\mathcal{P}(L_w^s) = 0.125|L_w^s|$ for all sets L_w^s. Examine all scenarios in an ascent order of $\mathcal{P}(L_w^s)$, i.e., an increasing cardinality of L_w^s, for all $w = 1, 2, 3$. Allocate ϵ_1, ϵ_2, and ϵ_3 in

$$\begin{aligned}
&\mathcal{P}(L_1^7) = 0.25 > \epsilon_1 = 0.2 \geq \mathcal{P}(L_1^8) = 0.125 \ \Rightarrow\ s_1' = 7 \text{ for commodity 1;}\\
&\mathcal{P}(L_2^3) = 0.5 > \epsilon_2 = 0.4 \geq \mathcal{P}(L_2^6) = 0.375 \ \Rightarrow\ s_2' = 3 \text{ for commodity 2;}\\
&\mathcal{P}(L_3^3) = 0.375 > \epsilon_3 = 0.3 \geq \mathcal{P}(L_3^2) = 0.25 \ \Rightarrow\ s_3' = 3 \text{ for commodity 3.}
\end{aligned}$$

Now transform the three chance constraints into deterministic inequalities as follows.

$$\begin{aligned}
y_{241} + y_{341} &\geq \xi_{41}^7 = 9\\
y_{242} + y_{342} &\geq \xi_{42}^3 = 5\\
y_{243} + y_{343} &\geq \xi_{43}^3 = 8.
\end{aligned}$$

Alternatively, one can formulate the problem as [Model1-MIP] by defining binary variables z_{4w}^s as whether the w^{th} probabilistic constraint violates scenario s at the destination 4. According to Definition 2, $\{L_1^7\}$, $\{L_2^3\}$, and $\{L_3^3\}$ are minimal cover sets for each commodity-based chance constraint, respectively. Generating the corresponding cover inequalities will lead to the same solution:

$$z^7_{41} \le 0 \Rightarrow d_{41} \ge \xi^7_{41} = 9,\ z^3_{42} \le 0 \Rightarrow d_{42} \ge \xi^3_{42} = 5,\ \text{and}\ z^3_{43} \le 0 \Rightarrow d_{43} \ge \xi^3_{43} = 8,$$

The original problem is then equivalent to solving three shortest path problems, which aim to transport deterministic demands (i.e., 9, 5, 8 units of commodities $1, 2$, and 3, respectively) from a single origin to node 4, with $(c_{ij} + a_{ijw})$ being the traveling costs of all $(i, j) \in A$ for commodity w. Given Paths 0–2–4, 1–3–4, and 2–4 being the shortest paths, an optimal solution is $x^*_{02} = 9$, $x^*_{24} = 9 + 8 = 17$, $x^*_{13} = x^*_{34} = 5$, yielding the minimum cost of 61.4.

4.2.3 Polynomial time algorithms for special cases of PNDP-cont-c/n

The following discussion of special cases and the development of algorithms can be applied to both PNDP-cont-c and PNDP-cont-c/n. Using PNDP-cont-c as an example, when $|D_w| = 1$ for all $w \in W$, we then have a single-row chance constraint for every commodity w. The special case corresponds to real-world applications in which only one destination exists for each commodity, who can be satisfied by one or multiple suppliers. Recall that with single-row chance constraints, we can reformulate PNDP-cont-c as a deterministic problem via the enumeration of violation risk thresholds, for all commodities $w \in W$.

We apply similar approaches as Algorithm 1 by considering one destination i_w for each commodity w, and revise the for-loop from Step 3 through Step 6 as follows. With respect to each $w \in W$, identify scenario s', such that $\mathcal{P}(L^{s'}_w) > \epsilon_w \ge \mathcal{P}(L^{s'+1}_w)$, and replace the chance constraint (5) with

$$\sum_{j:(j,i)\in A} y_{ji_w w} - \sum_{j:(i,j)\in A} y_{i_w jw} \ge \xi^{s'}_{i_w w} \quad \forall w \in W. \tag{24}$$

PNDP-cont-c is then equivalent to solving

$$\min\Big\{(2a) : (2b)\text{–}(2d), \text{ and } (24)\Big\},$$

which is further equivalent to

$$\min\left\{\sum_{w\in W}\sum_{(i,j)\in A} (c_{ij} + a_{ijw})y_{ijw} : (2c), (2d), \text{ and } (24)\right\}. \tag{25}$$

Moreover, the algorithm employs the shortest-path algorithm rather than solve W minimum cost flow problems, because of the single-demand-node assumption for each commodity. We elaborate special cases of "one supply, one demand" and "multiple supplies, one demand" as follows.

Case I: $|O_w| = |D_w| = 1,\ \forall w$: The algorithm seeks a shortest paths for each origin and destination pair, and solves W shortest-path problems, by setting arc cost as $(c_{ij} + a_{ijw})$ for all $(i, j) \in A$ and $w \in W$. The Dijkstra algorithm finds each shortest path in $O(|N|^2)$ by noting that all revised costs are nonnegative. For every commodity w, $\xi^{s'}_{i_w w}$ units of flow are transported on arcs of the corresponding path, yielding the optimal objective value of Formulation (25) as

$$\sum_{w \in W} \xi^{s'}_{i_w w} \left(\sum_{(i,j) \in \mathbb{S}_w} (c_{ij} + a_{ijw}) \right),$$

where $\mathbb{S}_w$ is a shortest path identified for commodity w, $\forall w \in W$.

Case II: $|O_w| > 1$ and $|D_w| = 1$, $\forall w$: When having multiple suppliers, we first compute the shortest paths from each supplier in O_w to the destination i_w, and start from the "cheapest" supplier to flow as much as its capacity allows. We greedily repeat the "capacity-saturating" procedure for each ordered origin, until $\xi^{s'}_{i_w w}$ units of required demand are all satisfied. The details are given in Algorithm 2.

Algorithm 2 Solve Formulation (25) for Case II.

for all commodities $w \in W$ **do**
  for all nodes $j \in O_w$ **do**
    Use Dijkstra algorithm to find a shortest path from node j to the singleton destination i_w, denoted as $\mathbb{S}(j, i_w)$.
  end for
  Order all origins in O_w such that the shortest distances $\ell_{\mathbb{S}(1,i_w)} \leq \ell_{\mathbb{S}(2,i_w)} \leq \cdots \leq \ell_{\mathbb{S}(|O_w|,i_w)}$.
  For all $w \in W$, denote e_w as the demand excess (i.e., the current unsatisfied demand) and t as the supplier number under examination. Set $e_w = \xi^{s'}_{i_w w}$ and $t = 1$.
  while $e_w > 0$ **do**
    if $o_t \geq e_w$ **then**
      Flow e_w on the shortest path $\mathbb{S}(t, i_w)$.
      Let $e_w = 0$.
    else
      Flow o_t on the shortest path $\mathbb{S}(t, i_w)$.
      Let $e_w = e_w - o_t$; $t = t + 1$.
    end if
  end while
end for
return Optimal $\mathbf{y}^*_w$ as flows sent on all shortest paths, optimal $x^*_{ij} = \sum_{w \in W} y^*_{ijw}$, $\forall (i,j) \in A$, and the minimum cost $\mathbf{cx}^* + \mathbf{ay}^*$.

4.3 Relationship between the PNDP-cont models

Considering all general PNDP-cont models we have discussed so far, only PNDP-cont-nc can be quickly solved by Algorithm 1. For PNDP-cont-joint, PNDP-cont-c, and PNDP-cont-n, we need to solve a deterministic MIP model (involving binary $\mathbf{z}$). The cover inequalities can be used to potentially improve computational efficacy. Next we describe how to use optimal solutions of specially designed PNDP-cont-nc for computing upper bounds of the objectives of the other models. These bounds can be incorporated into branch-and-bound nodes, together with lower bounds obtained by solving LP relaxations of the original MIP models.

Consider PNDP-cont-joint, which contains a joint chance constraint with a risk tolerance ϵ. Assume that demand distributions are independent among all $w \in W$ and $i \in D_w$. Let E_{iw} represent the event of no demand loss at node $i \in D_w$ for commodity $w \in W$, i.e., $\sum_{j:(j,i)\in A} y_{jiw} - \sum_{j:(i,j)\in A} y_{ijw} \geq d_{iw}$ being true. We have

$$\Pr(E_{iw},\ \forall w \in W,\ i \in D_w) = \prod_{w\in W,\ i\in D_w} \Pr(E_{iw}). \tag{26}$$

Given risk tolerances ϵ_{iw} for all combinations of i and w, let $(\mathbf{x}^*, \mathbf{y}^*)$ be an optimal solution to PNDP-cont-nc. Note that

$$\prod_{w\in W,\ i\in D_w} \Pr(E_{iw}(\mathbf{x}^*, \mathbf{y}^*)) \geq \prod_{w\in W,\ i\in D_w} (1-\epsilon_{iw}) \geq 1 - \sum_{w\in W,\ i\in D_w} \epsilon_{iw}, \tag{27}$$

where the last inequality is due to $\epsilon_{iw} \in [0, 1)$ for all i and w. Based on (26) and (27), $(\mathbf{x}^*, \mathbf{y}^*)$ is a feasible solution to PNDP-cont-joint by letting $\epsilon = \sum_{w\in W,\ i\in D_w} \epsilon_{iw}$. As a result, for PNDP-cont-joint with ϵ, one can design ϵ_{iw} in PNDP-cont-nc with $\sum_{w\in W,\ i\in D_w} \epsilon_{iw} = \epsilon$, to ensure the foregoing relationship holds. We then solve the designed PNDP-cont-nc in polynomial time to obtain a feasible solution to PNDP-cont-joint, which also yields an upper bound for the optimal objective. Tighter bounds can be generated by varying ϵ_{iw}, $\forall i$ and w, while making sure $\sum_{w\in W,\ i\in D_w} \epsilon_{iw} \geq \epsilon$.

Similarly, each joint chance constraint in PNDP-cont-c and PNDP-cont-n can be approximated by a series of single-row chance constraints in PNDP-cont-nc, while respectively ensuring

$$\epsilon_w = \sum_{i\in D_w} \epsilon_{iw}, \text{ and } \epsilon_i = \sum_{w\in W} \epsilon_{iw}. \tag{28}$$

An optimal solution to PNDP-cont-nc is then feasible to PNDP-cont-c and PNDP-cont-n, and provides a valid upper bound for the corresponding optimal objective values.

For computing the benchmark SNDPs described in this paper, we employ a decomposition algorithm, which follows standard Benders procedures. We demonstrate the details of decomposition and cutting-plane generation in Appendix A. The approach is alternative to solving the MIPs via off-the-shelf solvers, and can be also generalized for solving the deterministic MIP reformulations of the PNDP-bin-nc.

5. Numerical examples

We test our models and algorithms on two sets of numerical instances based on randomly generated networks and the Sioux-Falls network, respectively. Section 5.1 focuses on the comparison between joint and single-row chance constraints, as well as comparison of different algorithms for solving PNDP-cont, by testing moderate-size network instances. Section 5.2 computes representative models of PNDP and SNDP on instances of a real-world network. We aim to derive managerial insights of implementing various continuous/discrete PNDP and SNDP formulations under different demand/supply situations.

5.1 Randomly generated networks

5.1.1 Experimental setup

Here we generate random network instances that have sizes of $|N| = 20$ and 30, with a density (defined as $|A|$ divided by $|N| \times |N|$) being approximately 25%. Each instance has three commodities. We test $|\Omega| = 50, 100, 200$ scenarios, and let $|O_w| = 4$ and $|D_w| = 2$ for all commodities $w = 1, 2, 3$. We test the four PNDP-cont models with fixed flow variables. The aim is to compare QoS results yielded by different models and demonstrate the efficacy of implementing Algorithm 1 compared with directly solving the MIP models.

For all arcs (i, j), assign c_{ij} as the integer obtained from rounding up a random number generated from a uniform distribution over the interval $(0, 8]$. For all arcs $(i, j) \in A$, we randomly generate a_{ijw} from uniform distributions over intervals $[0.1, 0.3]$, $[0.1, 0.4]$, and $[0.2, 0.4]$ for w =1, 2, and 3, respectively. For every scenario $s \in \Omega$, we first generate a random number from $1, \dots, |\Omega|$. The probability p^s is then computed by dividing the random number by the sum of all random numbers generated for each scenario, such that $\sum_{s\in\Omega} p^s = 1$ is enforced. We randomly select nodes in N to be origins in O_w or destinations in D_w for all $w \in W$. Finally, for $w = 1, 2, 3$ we generate the amounts of supply/demand from uniform distributions depicted in Table 1. The designated

Table 1: Distributions for generating supply/demand at corresponding locations.

Commodity	Supply 1, 2	Supply 3, 4	Demand 1, 2
$w = 1$	U(20,30)	U(40,50)	U(50,60)
$w = 2$	U(10,15)	U(20,25)	U(25,30)
$w = 3$	U(5,7)	U(10,12)	U(12,15)

distributions also guarantee that $\sum_{i\in O_w} o_{iw} \geq \sum_{j\in D_w} d_{jw}$, $\forall w \in W$. Thus, given any possible data realizations from Table 1, we always have feasible solutions.

All models and algorithms use CPLEX 12.3 via ILOG Concert Technology with C++, and computations are performed on a HP Workstation z400 Windows 7 machine with Intel(R) Xeon(R) CPU E31230 3.20 GHz, and 8GB memory. The CPU time is reported in seconds.

5.1.2 Result summary

We test models of PNDP-cont-joint/nc to demonstrate the relationship between joint and single-row chance-constraints. For the single-row chance constraints for $w = 1, 2, 3$, we assume two cases having homogeneous (denoted by "-ho") and heterogeneous (denoted by "-he") risk tolerances among different chance constraints. Corresponding to each case of ϵ, the homogeneous case uses $\epsilon_{11} = \epsilon_{12} = \epsilon_{21} = \epsilon_{22} = \epsilon_{31} = \epsilon_{32} = \epsilon/6$ when $|D_w| = 2$. When $|D_w| = 1$, $\forall w = 1, 2, 3$, the heterogeneous case uses $\epsilon_{11} = \epsilon_{12} = 1/12, \epsilon_{21} = \epsilon_{22} = 1/6$, and $\epsilon_{31} = \epsilon_{32} = 1/4$. The later setting follows an intuition that we aim to guarantee higher QoS levels for satisfying demands with higher variations (as indicated in Table 1). For each combination, we test fifteen instances and compute the averages to report.

Table 2 and Table 3 respectively report the CPU time and optimal objective values of the three PNDP MIP models tested. We use ϵ = 0.03, 0.06, 0.15, and 0.3, indicated in Column ϵ, which guarantee the probability of no demand loss for any commodity at any node being no less than 97%, 94%, 85%, and 70%, respectively.

Table 2: CPU time of the PNDP chance-constrained models (in seconds)

	ϵ	PNDP-cont-joint			PNDP-cont-nc-ho			PNDP-cont-nc-he		
		$\|\Omega\| = 50$	$\|\Omega\| = 100$	$\|\Omega\| = 200$	$\|\Omega\| = 50$	$\|\Omega\| = 100$	$\|\Omega\| = 200$	$\|\Omega\| = 50$	$\|\Omega\| = 100$	$\|\Omega\| = 200$
$\|N\| = 20$	0.03	0.049	0.189	0.767	0.022	0.079	0.168	0.024	0.045	0.164
	0.06	0.170	0.395	3.207	0.022	0.127	0.472	0.019	0.069	0.472
	0.15	0.507	2.218	41.311	0.061	0.455	3.381	0.082	0.371	2.951
	0.3	1.179	7.046	1589.000	0.258	0.483	5.567	0.209	0.539	5.139
$\|N\| = 30$	0.03	0.048	0.250	0.873	0.028	0.123	0.266	0.029	0.076	0.212
	0.06	0.256	0.509	3.875	0.028	0.161	0.672	0.028	0.127	0.428
	0.15	0.595	2.646	91.265	0.045	0.567	4.947	0.144	0.446	4.096
	0.3	1.738	8.929	739.115	0.452	0.640	7.728	0.334	0.831	7.929

Table 3: Optimal objective values of the PNDP chance-constrained models

	ϵ	PNDP-cont-joint			PNDP-cont-nc-ho			PNDP-cont-nc-he		
		$\|\Omega\| = 50$	$\|\Omega\| = 100$	$\|\Omega\| = 200$	$\|\Omega\| = 50$	$\|\Omega\| = 100$	$\|\Omega\| = 200$	$\|\Omega\| = 50$	$\|\Omega\| = 100$	$\|\Omega\| = 200$
$\|N\| = 20$	0.03	781.93	766.87	868.07	781.93	766.87	868.07	781.93	766.87	868.07
	0.06	763.20	748.20	850.89	781.93	757.65	861.64	763.20	748.20	850.89
	0.15	674.73	659.81	761.93	724.65	691.79	784.59	674.73	659.81	761.93
	0.3	663.94	648.84	750.92	717.99	684.55	779.52	663.94	648.84	750.92
$\|N\| = 30$	0.03	682.07	692.80	650.47	682.07	693.13	650.47	682.07	692.80	650.47
	0.06	664.53	679.40	636.60	682.07	690.27	643.71	664.53	679.40	636.60
	0.15	574.60	586.74	545.13	664.32	619.01	571.46	574.60	586.74	545.13
	0.3	564.17	579.68	536.05	663.91	617.63	565.69	564.17	579.68	536.05

For PNDP-cont-nc, we solve all instances with Algorithm 1 and the MIP approach, and depict the CPU time comparison in Figure 2. The CPU time of the MIP approach on the heterogeneous 200-scenario instances is used as the benchmark, being the pair of approach and instances requiring the most CPU time. The CPU time of the other cases is then compared as a percentage of this benchmark.

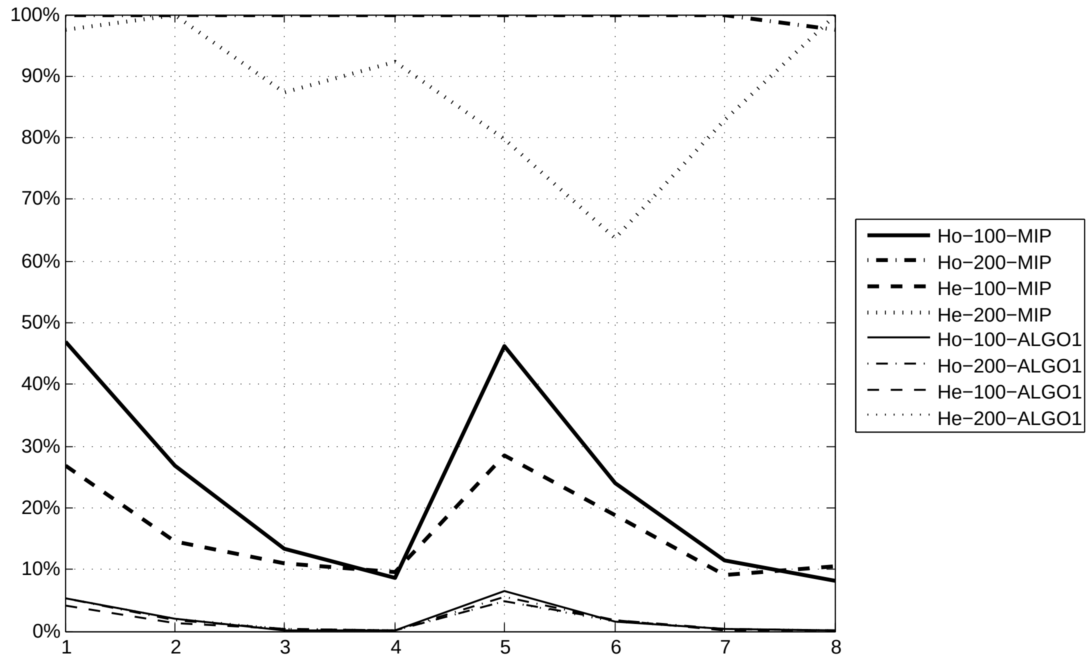


Figure 2: Percentage comparison of CPU time taken by Algorithm 1 and the MIP approach

We list our observations based on the computational results as follows.

- The optimal objective values decrease as we increase ϵ (i.e. by allowing more unmet demands and lower QoS levels). Also, PNDP-cont-nc is less sensitive to changes in ϵ compared to PNDP-cont-joint. This is because the risk is divided and distributed into several chance constraints, and thus the same change in ϵ will result in relatively smaller changes in ϵ_i for each constraint i.

- As described in Section 4.3, optimal solutions to PNDP-cont-nc will serve as feasible solutions to PNDP-cont-joint when $\epsilon \leq \sum_{w\in W, i\in D_w} \epsilon_{iw}$. Such (upper) bounds in general become tighter when ϵ is small. (In particular, when $\epsilon = 0.03$, optimal solutions to PNDP-cont-nc are also optimal to PNDP-cont-joint.) The bounds get much worse after we lower the QoS level from 94% to 85% (i.e., increasing ϵ from 0.06 to 0.15). The heterogeneous risk setting in general yields better bounds than the homogeneous case, indicating the importance and necessity of differentiating risk tolerances (or QoS levels) for different customers and commodities. In general, all bounds become tighter in both tables when $|\Omega|$ increases.

- For all MIP models tested, the CPU time dramatically increases as we increase ϵ (i.e. when allowing higher probabilities of violating chance constraints). We have much longer computational time spent on solving the MIPs compared with Algorithm 1, which also dramatically increases as we increase $|\Omega|$.

- Solving the MIP model of PNDP-cont-nc is significantly faster than solving the MIP model of PNDP-cont-joint. In particular, the CPU time taken by Algorithm 1 is almost the same for all instances, regardless of changes to (i) the number of scenarios $|\Omega|$, (ii) the sum of risk tolerances ϵ, and (iii) homogeneous or heterogeneous risk settings. This is consistent with the observation that the complexity of Algorithm 1 is not determined by the number of scenarios but by the complexity of solving $|W|$ minimum-cost-flow problems.

5.2 Sioux-Falls network

5.2.1 Experimental setup

In this experiment, we use the Sioux Falls road network (LeBlanc et al. 1975), as shown in Figure 3, which is widely used in transportation literature. This network consists of 24 nodes and 76 links. We continue to use three commodities and $|\Omega| = 100$ scenarios in instances created based on this network. The aim in this experiment is to glean managerial insights through the use and comparison of PNDP and SNDP models on an instance that is closer to a real world instance.

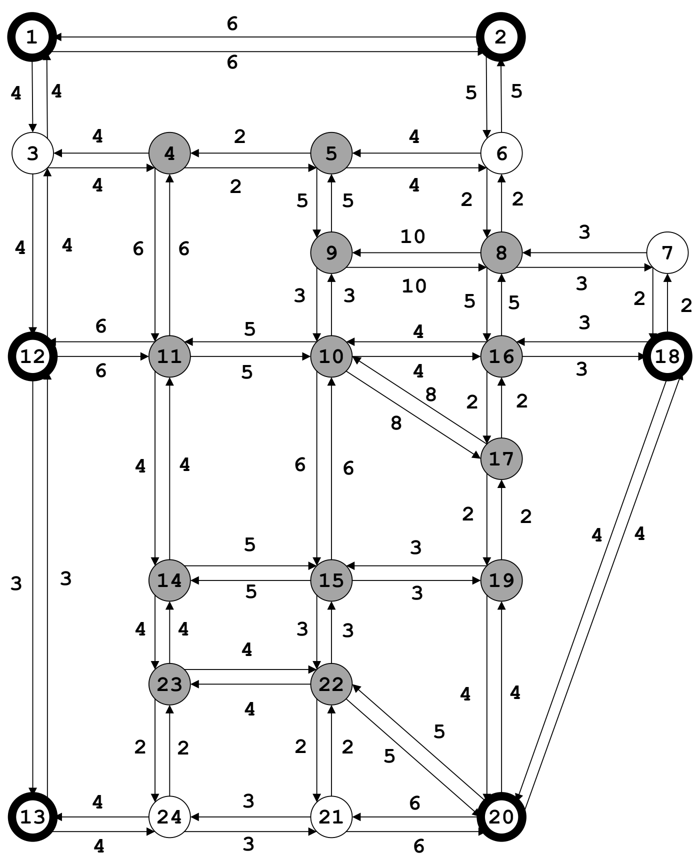


Figure 3: Sioux Falls road network

We simulate a high inflow instance, where commodities flow into the network exclusively from the outer nodes of the network, with higher mean demands for nodes that are more centralized in the network. We select nodes 1, 2, 12, 13, 18, and 20 as the origins (bold nodes in Figure 3), as these are the most likely entry points into Sioux Falls, and select the inner nodes 4, 5, 8–11, 14–17, 19, 22, and 23 as the destinations (shaded nodes), as these are the more populated areas in Sioux Falls.

For all arcs (i, j), the travel distances c_{ij} between nodes are indicated on the arcs in Figure 3. For simplicity, the arcs are symmetric, i.e. (j, i) always exists and is always equal in length to (i, j), for all $(i, j) \in A$. For each arc (i, j) and commodity $w \in W$, we assign a_{ijw} as 0.2, 0.25, and 0.3 for $w = 1, 2, 3$ respectively. For each scenario $s \in \Omega$, we generate the probability p^s by following the same method in Section 5.1.

Denote the mean demands for commodity w at node j by $\bar{d}_{jw}$. We set the mean demands at node 10 (the *center*) to be $\bar{d}_{10,1} = 1000, \bar{d}_{10,2} = 3000$, and $\bar{d}_{10,3} = 7000$, and set the mean demands at the other destination nodes j to be $\bar{d}_{jw} = (1 - \lambda^k)\bar{d}_{10,w}$, $\forall w = 1, 2, 3$, where k is the minimum number of links of a path from node 10 to node j and λ is the average rate of decay of demand from the center. Finally, we sample the realizations ξ^s_{jw} of d_{jw} from the distribution $U(0, 2\bar{d}_{jw})$, $w = 1, 2, 3$.

For each of the two instances, the supply for each commodity at each origin is then generated randomly, while ensuring that $\sum_{i \in O_w} o_{iw} \geq \sum_{j \in D_w} \xi^s_{jw}$, $\forall w \in W,\ s \in \Omega$, so that we will always have feasible solutions in every scenario.

5.2.2 Results summary

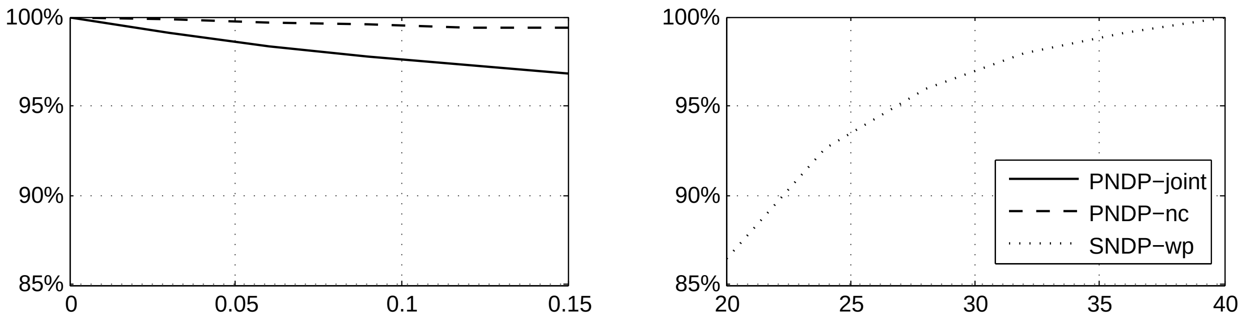
We analyze the sensitivity of optimal objective values to the parameters ϵ and v for different values of λ. The models of PNDP-joint, PNDP-nc, and SNDP-wp are used in this comparison. Here, we used the homogenous versions of PNDP-nc (i.e. $\epsilon_{iw} = \epsilon/39$ for each constraint iw) and SNDP-wp (i.e., $v_{iw} = v$ for each violation of constraint i-w).

For the PNDP-joint and PNDP-nc cases, we used ϵ from 0 to 0.15, in intervals of 0.03. For the SNDP-wp case, we used v from 20 to 40, in intervals of 4. For each of these values of ϵ or v, the optimal values of each model was found and taken as a percentage of the optimal value of the most restricted instance of each model, i.e. $\epsilon = 0$ for both PNDP models, and $v = 40$ for the SNDP-wp model. Figure 4 illustrates the comparison of the three models for different levels of demand decay.

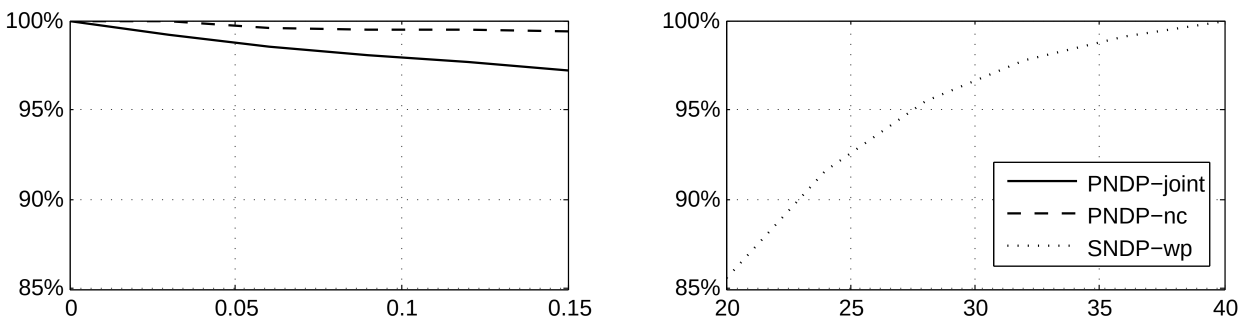
We list our observations based on the computational results as follows.

- The optimal values of PNDP-joint and PNDP-nc have a fairly linear relationship with ϵ. This is in contrast with the curved graph of the SNDP-wp model; this is to be expected. As v increases, the dominant term in the bi-objective function changes from the real cost of the solution to the virtual penalty cost incurred by unmet demand.

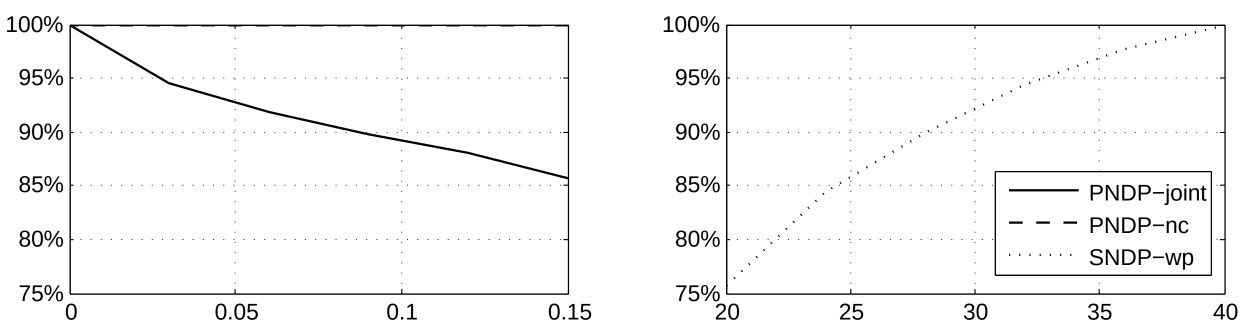
- The above point reveals the rather ambiguous nature of the penalty value v - without first experimenting with several values of v, one cannot determine a suitable value of v for which the objective function weighs the actual and virtual costs in a manner that is reasonable to the decision maker. PNDP models in general mitigate the ambiguity of solution reliability,



(a) $\lambda = 0$: even distribution of demand among destination nodes.



(b) $\lambda = 0.2$: moderate decay of demand from node 10.



(c) $\lambda = 1$: all demand at node 10 only.

Figure 4: Comparisons of the objective values of SNDP-wp, PNDP-joint, and PNDP-nc.

and provide a decision maker with confidence levels on the QoS that (s)he can place in the solution, which an SNDP model cannot usually achieve.

- The optimal objectives of PNDP-cont-nc serve relatively good upper bounds for the optimal objectives of PNDP-cont-joint, and become tighter when (i) $|\Omega|$ increases, and/or (ii) in the heterogeneous risk setting. This might be of interest to some decision makers, who try to satisfy certain QoS levels by prioritizing their customer demands and requiring higher QoS levels for demands with higher variations. Instead of computing a time-consuming MIP of PNDP-cont-joint with a joint chance constraint, one can solve PNDP-cont-nc as a variant with multiple single-row chance constraints. This approximation in general might provide very tight bounds and high quality solutions based on all numerical results.

6. Conclusions and future research

In this paper, we analyzed model variants and solution approaches of the probabilistic multi-commodity flow capacity design problem. We first examined PNDP-cont-joint with a joint chance constraint that guarantees certain probability of no-demand losses at all nodes for all commodities. The problem was reformulated as an MIP by defining binary variables associated with each scenario. We then formulated three model variants that distribute risks into multiple chance constraints, namely, PNDP-cont-nc, PNDP-cont-c, and PNDP-cont-n. In addition to MIP reformulations, we discussed polynomial algorithms for solving PNDP-cont-nc by identifying risk thresholds of every single-row chance constraint. The modified approach then transformed the problem, and solved several shortest-path problems to attain optimality. Similar approaches were developed for solving special cases of PNDP-cont-c and PNDP-cont-n. We formulated benchmark stochastic programming models by either enforcing to meet all demand or penalizing unmet demand via a linear cost function, and tested different models and approaches on randomly generated network instances and an instance given by the Sioux-Falls network. Our results show that differentiating QoS levels for different commodities and/or customers (i.e., using models with multiple chance constraints) can result in cost savings in network capacity design and transportation, as well as can yield better solution bounds (with much shorter computational times needed) for models having joint chance constraints (i.e., unified QoS levels).

In our future research, we will continue to examine risk correlations among multiple chance constraints. One way is to consider risk tolerances as decision variables, and to seek an optimal combination of risk and cost. We are also interested in investigating special network topologies to derive effective algorithms for optimizing both PNDP and SNDP models.

Acknowledgments. The authors are grateful to all anonymous referees and the editor-in-chief, whose remarks led to an improved contribution and presentation of the paper. Dr. Shen also gratefully acknowledges the support of IBM Smarter Planet Innovation Faculty Award.

References

Ahuja, R. K., Magnanti, T. L., Orlin, J. B., 1993. Network Flows: Theory, Algorithms, and Applications. Prentice Hall, Upper Saddle River, NJ.

Barton, R. R., Hearn, D. W., Lawphongpanich, S., 1989. The equivalence of transfer and generalized benders decomposition methods for traffic assignment. Transportation Research Part B: Methodological 23 (1), 61–73.

Chang, M.-S., Tseng, Y.-L., Chen, J.-W., 2007. A scenario planning approach for the flood emergency logistics preparation problem under uncertainty. Transportation Research Part E: Logistics and Transportation Review 43 (6), 737–754.

Charnes, A., Cooper, W., Symonds, G., 1958. Cost horizons and certainty equivalents: An approach to stochastic programming of heating oil. Management Science 4 (3), 235–263.

Chen, A., Yang, C., 2004. Stochastic transportation network design problem with spatial equity constraint. Transportation Research Record: Journal of the Transportation Research Board 1882 (-1), 97–104.

Chen, A., Zhou, Z., Chootinan, P., Ryu, S., Yang, C., Wong, S., 2011. Transport network design problem under uncertainty: a review and new developments. Transport Reviews 31 (6), 743–768.

Chung, B. D., Yao, T., Xie, C., Thorsen, A., 2011. Robust optimization model for a dynamic network design problem under demand uncertainty. Networks and Spatial Economics 11 (2), 371–389.

Gao, Z., Wu, J., Sun, H., 2005. Solution algorithm for the bi-level discrete network design problem. Transportation Research Part B: Methodological 39 (6), 479–495.

Kulshrestha, A., Wu, D., Lou, Y., Yin, Y., 2011. Robust shelter locations for evacuation planning with demand uncertainty. Journal of Transportation Safety & Security 3 (4), 272–288.

LeBlanc, L. J., Morlok, E. K., Pierskalla, W. P., 1975. An efficient approach to solving the road network equilibrium traffic assignment problem. Transportation Research 9 (5), 309–318.

Li, C., Yang, H., Zhu, D., Meng, Q., 2012. A global optimization method for continuous network design problems. Transportation Research Part B: Methodological 46 (9), 1144–1158.

Lo, H. K., Tung, Y.-K., 2003. Network with degradable links: capacity analysis and design. Transportation Research Part B: Methodological 37 (4), 345–363.

Lou, Y., Yin, Y., Lawphongpanich, S., 2009. Robust approach to discrete network designs with demand uncertainty. Transportation Research Record: Journal of the Transportation Research Board 2090 (-1), 86–94.

Luathep, P., Sumalee, A., Lam, W. H., Li, Z.-C., Lo, H. K., 2011. Global optimization method for mixed transportation network design problem: A mixed-integer linear programming approach. Transportation Research Part B: Methodological 45 (5), 808–827.

MirHassani, S., Lucas, C., Mitra, G., Messina, E., Poojari, C., 2000. Computational solution of capacity planning models under uncertainty. Parallel Computing 26 (5), 511–538.

Nemhauser, G. L., Wolsey, L. A., 1988. Integer and Combinatorial Optimization. Wiley, New York.

Oh, S.-C., Haghani, A., 1997. Testing and evaluation of a multi-commodity multi-modal network flow model for disaster relief management. Journal of Advanced Transportation 31 (3), 249–282.

Patil, G. R., Ukkusuri, S. V., 2007. System-optimal stochastic transportation network design. Transportation Research Record: Journal of the Transportation Research Board 2029 (-1), 80–86.

Patriksson, P., 1994. The traffic assignment problem: Models and methods. Topics in Transportation Series. VSP Intl Science, Utrecht, The Netherlands.

Ruszczyński, A., 2002. Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra. Mathematical Programming 93 (2), 195–215.

Santoso, T., Ahmed, S., Goetschalckx, M., Shapiro, A., 2005. A stochastic programming approach for supply chain network design under uncertainty. European Journal of Operational Research 167 (1), 96–115.

Sharma, S., Mathew, T. V., Ukkusuri, S. V., 2010. Approximation techniques for transportation network design problem under demand uncertainty. Journal of Computing in Civil Engineering 25 (4), 316–329.

Sheu, J., 2007. Challenges of emergency logistics management. Transportation Research Part E: Logistics and Transportation Review 43 (6).

Ukkusuri, S. V., Mathew, T. V., Waller, S. T., 2006. Robust transportation network design under demand uncertainty. Computer-Aided Civil and Infrastructure Engineering 22 (1), 6–18.

Ukkusuri, S. V., Patil, G., 2009. Multi-period transportation network design under demand uncertainty. Transportation Research Part B: Methodological 43 (6), 625–642.

Ukkusuri, S. V., Waller, S. T., 2008. Linear programming models for the user and system optimal dynamic network design problem: formulations, comparisons and extensions. Networks and Spatial Economics 8 (4), 383–406.

Waller, S. T., Ziliaskopoulos, A. K., 2001. Stochastic dynamic network design problem. Transportation Research Record: Journal of the Transportation Research Board 1771 (-1), 106–113.

Wang, S., Meng, Q., Yang, H., 2013. Global optimization methods for the discrete network design problem. Transportation Research Part B: Methodological 50, 42–60.

Wolsey, L. A., 1998. Integer Programming. Wiley, New York, NY.

Yin, Y., Madanat, S. M., Lu, X. Y., 2009. Robust improvement schemes for road networks under demand uncertainty. European Journal of Operational Research 198 (2), 470–479.

APPENDIX

A. A Benders decomposition approach for all SNDPs

We note that SNDPs require that their flow variables for each scenario be optimal for that particular scenario, but also require the capacity design to be common across all scenarios. This nature of SNDPs lends itself nicely to a Benders decomposition approach when we optimize SNDPs. We use SNDP-bin-wp to illustrate the use of Benders decomposition as it is the most complex of the four SNDP variants. However, this approach can be used for all variants of SNDP.

Let the capacity design variables β be the first-stage variables, and the recourse flow variables $\mathbf{y}^s$ and variables representing unmet demand $\mathbf{t}^s$ be second stage variables. In the *master problem* **[MP]**, we optimize the objective over relaxed constraints on β and on θ, the lower bounds on the subproblem in each scenario. At each iteration, **[MP]** is solved to obtain a trial solution (β, θ), which is passed to the subproblems described later. Initially, the feasibility and optimality cut sets $L_1(\beta) \geq 0$ and $L_2(\beta, \theta) \geq 0$ have no cuts.

[MP]:

$$\min \quad \sum_{(i,j)\in A} q_{ij}\beta_{ij} + \sum_{s\in\Omega} p^s\theta_s \tag{A-1a}$$

$$\text{s.t.} \quad L_1(\beta) \geq 0 \tag{A-1b}$$

$$L_2(\beta, \theta_1, \ldots, \theta_{|\Omega|}) \geq 0 \tag{A-1c}$$

$$\beta_{ij} \in \{0, 1\} \quad \forall (i,j) \in A \tag{A-1d}$$

We denote the *subproblems* by $[\mathbf{SP}(s, \beta)]$. For each scenario s, the trial solution obtained from $[\mathbf{MP}]$ is used as a parameter to obtain an optimal $(\mathbf{y}^s, \mathbf{t}^s)$ through $[\mathbf{SP}(s, \beta)]$. A feasibility or optimality cut is generated depending on whether $[\mathbf{SP}(s, \beta)]$ is infeasible, or has an optimal value that is greater than θ_s, and the cut is appended to the appropriate cut set. (Problem $[\mathbf{SP}(s, \beta)]$ is never unbounded since its optimal value is nonnegative.) If the optimal value for $[\mathbf{SP}(s, \beta)]$ is at most θ_s, then the solution on hand is declared to be optimal for SNDP-bin-wp.

We note that while $[\mathbf{SP}(s, \beta)]$ is not required in the algorithm to generate cuts, we provide its formulation for the sake of completeness.

$[\mathbf{SP}(s, \beta)]$:

$$\min \quad \sum_{w \in W} \left(\sum_{(i,j) \in A} a_{ijw} y^s_{ijw} + \sum_{i \in D_w} v_{iw} t^s_{iw} \right) \tag{A-2a}$$

$$\text{s.t.} \quad \sum_{w \in W} y^s_{ijw} \leq u_{ij} \beta_{ij} \quad \forall (i,j) \in A \tag{A-2b}$$

$$\sum_{j:(i,j) \in A} y^s_{ijw} - \sum_{j:(j,i) \in A} y^s_{jiw} \leq o_{iw} \quad \forall i \in O_w,\ w \in W \tag{A-2c}$$

$$\sum_{j:(i,j) \in A} y^s_{ijw} - \sum_{j:(j,i) \in A} y^s_{jiw} = 0 \quad \forall i \notin O_w \cup D_w,\ w \in W \tag{A-2d}$$

$$- \sum_{j:(i,j) \in A} y^s_{ijw} + \sum_{j:(j,i) \in A} y^s_{jiw} + t^s_{iw} \geq \xi^s_{iw} \quad \forall i \in D_w,\ w \in W \tag{A-2e}$$

$$\mathbf{y}^s \geq 0,\ \mathbf{t}^s \geq 0 \tag{A-2f}$$

To determine whether $[\mathbf{SP}(s, \beta)]$ is feasible, we determine whether the *dual* $[\mathbf{D\text{-}SP}(s, \beta)]$ of the subproblem is unbounded.

$[\mathbf{D\text{-}SP}(s, \beta)]$:

$$\max \quad \sum_{(i,j) \in A} (u_{ij} \beta_{ij}) \mu^s_{ij} + \sum_{w \in W} \left(\sum_{i \in O_w} o_w \pi^s_{iw} - \sum_{i \in D_w} \xi^s_{iw} \pi^s_{iw} \right) \tag{A-3a}$$

$$\text{s.t.} \quad \mu^s_{ij} + \pi^s_{iw} - \pi^s_{jw} \leq a_{ijw} \quad \forall (i,j) \in A,\ w \in W \tag{A-3b}$$

$$\mu^s_{ij} \leq 0 \quad \forall (i,j) \in A \tag{A-3c}$$

$$\pi^s_{iw} \leq 0 \quad \forall i \in O_w,\ w \in W \tag{A-3d}$$

$$\pi^s_{iw} \leq v_{iw} \quad \forall i \in D_w,\ w \in W \tag{A-3e}$$

where μ^s_{ij} and π^s_{iw} are the dual variables associated with the constraints (A-2b) and (A-2c)–(A-2e) respectively. In other words, we determine if there exists an unbounded dual direction. To do this, we solve the *separation problem* $[\mathbf{S\text{-}SP}(s, \beta)]$.

[S-SP(s,β)]:

$$\begin{aligned}
\max \quad & \sum_{(i,j)\in A} (u_{ij}\beta_{ij})\mu^s_{ij} + \sum_{w\in W}\left(\sum_{i\in O_w} o_w\pi^s_{iw} - \sum_{i\in D_w}\xi^s_{iw}\pi^s_{iw}\right) \\
\text{s.t.} \quad & \mu^s_{ij} + \pi^s_{iw} - \pi^s_{jw} \le 0 \quad \forall (i,j)\in A,\ w\in W \\
& \mu^s_{ij} \le 0 \quad \forall (i,j)\in A \\
& \pi^s_{iw} \le 0 \quad \forall i \in O_w \cup D_w,\ w\in W
\end{aligned}$$

Its feasible region consists of vectors that, when added to feasible solutions of **[D-SP(s,β)]**, does not change their feasibility in **[D-SP(s,β)]**. If its objective function is positive, then an unbounded dual direction exists, and the optimal solution in this case would be an unbounded dual direction of **[D-SP(s,β)]**. Algorithm 3 describes the algorithm in greater detail.

Remark 2. It can be observed that Algorithm 3 can easily be modified to solve the other three variants of SNDP by removing the penalty term in the case without penalty, or by changing the binary design variables to continuous design variables. For either modification (or both together), the second stage is still a linear program, so Benders decomposition remains a viable algorithm to solve the problem.

The Benders approach can also be applied to optimize PNDP-bin-nc. As mentioned in Section 3.3, replacing the continuous variables $\mathbf{x}$ with the binary variables β increases the difficulty of the problem. To reduce the difficulty in solving PNDP-bin-nc, we can decompose the problem instead of solving the MIP directly, again with the network design variables as the first stage variables and the flow variables as the second stage variables, resulting in a linear program in the second stage. The above methodology can then be applied appropriately, with careful consideration to the fact that there is only one subproblem when decomposing PNDP-bin-nc, instead of the $|\Omega|$ subproblems when decomposing SNDP-bin-wp in Algorithm 3.

Remark 3. It is also important to note that **[SP(s,β)]** as stated above in (A-2) is, in fact, always feasible, because there always exists the feasible solution of having zero flow and all demands unsatisfied. This is also true of the subproblems of SNDP-cont-wp. However, the feasibility check is included in Algorithm 3 to illustrate a more general Benders approach that can be applied to the other problems mentioned in Remark 2.

Algorithm 3 (Generalized) Benders decomposition algorithm for SNDP-bin-wp.

Initialize the iteration number $t = 0$, and initialize [**MP**] without any cuts in $L_1(\beta) \geq 0$ and $L_2(\beta, \theta) \geq 0$.

Solve [**MP**] and obtain an optimal solution (β^t, θ^t). If there are no cuts in $L_2(\beta, \theta) \geq 0$, let $\theta_s^t = -\infty$ for all $s \in \Omega$.

repeat

  Increment t by 1.

  for all $s \in \Omega$ **do**

    Feasibility check

    Solve [**S-SP**(s, β^t)].

    if the optimal value of [**S-SP**(s, β^t)] is positive **then**

      Let $(\mu^{s,t}, \pi^{s,t})$ be the optimal solution of [**S-SP**(s, β^t)], and generate the cut

$$\sum_{(i,j)\in A} (u_{ij}\mu_{ij}^{s,t})\beta_{ij} + \sum_{w\in W} \left(\sum_{i\in O_w} o_w \pi_{iw}^{s,t} - \sum_{i\in D_w} \xi_{iw}^s \pi_{iw}^{s,t} \right) \leq 0$$

      into the cut set $L_1(\beta) \geq 0$ in [**MP**].

    else

      Optimality check

      Solve [**D-SP**(s, β^t)].

      if the optimal value of [**D-SP**(s, β^t)] $> \theta_s^t$ **then**

        Let $(\mu^{s,t}, \pi^{s,t})$ be the optimal solution of [**D-SP**(s, β^t)], and generate the cut

$$\sum_{(i,j)\in A} (u_{ij}\mu_{ij}^{s,t})\beta_{ij} + \sum_{w\in W} \left(\sum_{i\in O_w} o_w \pi_{iw}^{s,t} - \sum_{i\in D_w} \xi_{iw}^s \pi_{iw}^{s,t} \right) \leq \theta_s$$

        into the cut set $L_2(\beta, \theta) \geq 0$ in [**MP**].

      end if

    end if

  end for

until No cuts were added in iteration t.

Claim optimality of (β^t, θ^t).