Nurse Staffing under Absenteeism: A Distributionally Robust Optimization Approach

Minseok Ryu, Ruiwei Jiang
Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, MI 48109
Email: {msryu,ruiwei}@umich.edu

Abstract

We study the nurse staffing problem under random nurse demand and absenteeism. While the demand uncertainty is exogenous (stemming from the random patient census), the absenteeism uncertainty is *endogenous*, i.e., the number of nurses who show up for work partially depends on the nurse staffing level. For the quality of care, many hospitals have developed float pools of nurses by cross-training, so that a pool nurse can be assigned to the units short of nurses. In this paper, we propose a distributionally robust nurse staffing (DRNS) model that considers both exogenous and endogenous uncertainties. We derive a separation algorithm to solve this model under an arbitrary structure of float pools. In addition, we identify several pool structures that often arise in practice and recast the corresponding DRNS model as a monolithic mixed-integer linear program, which facilitates off-the-shelf commercial solvers. Furthermore, we optimize the float pool design to reduce the cross-training while achieving a specified target staffing costs. The numerical case studies, based on the data of a collaborating hospital, suggest that the units with high absenteeism probability should be pooled together.

Keywords: Nurse staffing; endogenous uncertainty; distributionally robust optimization; strong valid inequalities; convex hull

1 Introduction

Nurse staffing plays a key role in hospital management. The cost of staffing nurses accounts for over 30% of the overall hospital annual expenditures (see, e.g., [43]). Besides, the nurse staffing level make significant impacts on patient safety, quality of care, and the job satisfaction of nurses (see, e.g., [38]). In view of that, a number of governing agencies (e.g., the California Department of Health [8] and the Victoria Department of Health [33]) have set up minimum nurse-to-patient ratios (NPRs) for various types of hospital units to regulate the staffing decision.

In general, the nurse planning consists of the following four phases: (1) nurse demand forecasting and staffing, (2) nurse shift scheduling, (3) pre-shift staffing and re-scheduling, and (4) nurse-patient assignment (see [3, 18, 25, 2]). In particular, phase (1) takes place weeks or months ahead of a shift and determines the nurse staffing levels based on, e.g., the forecasted patient census and the NPRs; and phase (3) takes place hours before the shift and recruits additional workforce (e.g., temporary or off-duty nurses) if any units are short of nurses. In this paper, we focus on these two phases and refer the corresponding decision making process as nurse staffing. The outputs of our study (e.g., the nurse staffing levels) can be used in phases (2) and (4) to generate shift schedules and assignments of the nurses.

Nurse staffing is a challenging task, largely because of the uncertainties of nurse demand and absenteeism. The demand uncertainty stems from the random patient census and has been well documented (see, e.g., [15, 6]) and studied in the nurse staffing literature (see, e.g., [12, 23]). In contrast, the absenteeism uncertainty has received relatively less attention in this literature (see, e.g., [16, 28]), albeit commonly observed in practice. For example, according to the U.S. Bureau of Labor Statistics [34], the average absence rate among all nurses in the Veterans Affairs Health Care System is 6.4% [41], significantly higher than that among all occupations (2.9%) and among health-care support occupations (4.3%). For the quality of care, many hospitals have developed float pools of nurses by cross-training, so that in phase (3) a pool nurse can be assigned to the units short of nurses (see, e.g., [19]).

Unlike the demand, the random number of nurses who show up for a shift partially depends on the nurse staffing level, i.e., the absenteeism uncertainty is *endogenous*. For example, if the nurse staffing level is $w \in \mathbb{N}_+$ then the random number of nurses who show up cannot exceed w. Although failing to incorporate such endogeneity may result in understaffing (see [16]), unfortunately, modeling endogeneity usually makes optimization models computationally prohibitive (see, e.g., [13]). Due to this technical difficulty, the endogenous uncertainty has received much less attention in the literature of stochastic optimization than the exogenous uncertainty. Existing works often resort to exogenous uncertainty for an approximate solution. Alternatively, they employ certain parametric probability distributions to model the endogenous uncertainty (see [13]), e.g., the absence of each nurse follows *independent* Bernoulli distribution with the *same* probability (which may depend on the staffing level; see [16]). A basic challenge to adopting parametric models is that a complete and accurate knowledge of the endogenous probability distribution is usually unavailable. Under many circumstances, we only have historical data, including the nurse staffing level and the corresponding absence records, which can be considered as samples taken from the true (but ambiguous) endogenous distribution. As a result, the solution obtained by assuming a parametric model can yield unpleasant out-of-sample performance if the chosen model is biased.

In this paper, we propose an alternative, nonparametric model of both exogenous and endogenous uncertainties based on distributionally robust optimization (DRO). Our approach considers a family of probability distributions, termed an ambiguity set, based only on the support and moment information of these uncertainties. In particular, the number of nurses who show up in a unit/pool is bounded by the corresponding staffing level and its mean value is a function of this level. Then, we employ this ambiguity set in a two-stage distributionally robust nurse staffing (DRNS) model that imitates the decision making process in phases (1) and (3). Building on DRNS, we further search for *sparse* pool structures that result in a minimum amount of cross-training while achieving a specified target staffing cost. To the best of our knowledge, this is the first study of the endogenous uncertainty in nurse staffing by using a DRO approach.

1.1 Literature Review

A vast majority of the nurse staffing literature focuses on deterministic models that do not take into account the randomness of the nurse demand and/or absenteeism (see [40]). Various (deterministic) optimization models have been employed, including linear programming (see, e.g., [22, 7]) and mixed-integer programming (see, e.g., [39, 44, 29, 2]). For example, [22] assessed the need for hiring permanent staffs and temporary helpers and [39] analyzed the trade-offs among hiring full-time, part-time, and overtime nurses. More recently, [44] compared cross-training and flexible work days and demonstrated that cross-training is far more effective for performance improvement than flexible work days. Similarly, [2] identified cross-training as a promising extension from their deterministic model. Despite the potential benefit of operational flexibility brought by float pools and cross-training, [29] pointed out that the pool design and staffing are often made manually in a qualitative fashion (also see [37]). In addition, when the nurse demand and/or absenteeism is random, the deterministic models may underestimate the total staffing cost (see, e.g., [21]).

Existing stochastic nurse staffing models often consider the demand uncertainty only. For example, [9] studied a two-stage stochastic programming model that integrates the staffing and scheduling of cross-trained workers (e.g., nurses) under demand uncertainty. Through numerical tests, [9] demonstrated that cross-training can be even more valuable than the perfect demand information (i.e., knowing the realization of demand when making staffing decisions). In addition, [26] studied how the mandatory overtime laws can negatively effect the service quality of a nursing home. Using a two-stage stochastic programming model under demand uncertainty, [26] pointed out that these laws result in a lower staffing level of permanent registered nurses and a higher staffing level of temporary registered nurses. Unfortunately, as [16] pointed out, ignoring nurse absenteeism may result in understaffing, which reduces the service quality and increases the operational cost because additional temporary nurses need to be called in.

When the nurse absenteeism is taken into account, the stochastic optimization models become unscalable. [16] considered the staffing of a single unit under both nurse demand and absenteeism uncertainty and successfully derived a closed-form optimal staffing level. In addition, [36] studied the staffing of a single on-call pool that serves multiple units whose staffing levels are fixed and known. In a setting that regular nurses can be absent while pool nurses always show up, the authors successfully derived a closed-form optimal pool staffing level. Unfortunately, the problem becomes computationally prohibitive when multiple units and/or multiple float pools are incorporated. For example, [14] studied a multi-unit and one-pool setting[1]. The author showed that the proposed stochastic optimization model outperforms the (deterministic) mean value approximation. However, the evaluation of this model "does not scale well." More specifically, even when staffing levels are *fixed*, one needs to solve an exponential number (in terms of the staffing level) of linear programs to evaluate the expected total cost of staffing. This renders the search of an optimal staffing level so challenging that one has to resort to heuristics. [41] considered a multi-unit and no-pool setting and analyzed the staffing problem based on a cohort of nurses who have heterogeneous absence rates. The authors showed that the staffing cost is lower when the nurses are heterogenous within each unit but uniform across units. Unfortunately, searching for an optimal staffing strategy is "computationally demanding" with a large number of nurses. Similar to [14], [41] resort to easy-to-use heuristics.

To mitigate the computational challenges of nurse absenteeism, the existing literature often make parametric assumptions on the endogenous probability distribution. For example, [16, 14, 41] assumed that the absences of all nurses are stochastically *independent* and the absence rate in [16, 14] is assumed *homogeneous*. But the nurse absences may be positively correlated during extreme weather (e.g., heavy snow) or during day shifts (e.g., due to conflicting family obligations). In addition, the data analytic in [41] suggests that the nurses actually have heterogeneous absence rates. Furthermore, the absenteeism can be drastically different among different units/hospitals, and even within the same unit/hospital, has high temporal variations. For example, based on the data from different hospitals, [16] concluded that the absence rate depends on the staffing level and ignoring such dependency results in understaffing, while [41] concluded that such dependency is insignificant. A fundamental challenge to adopting parametric models is that the solution thus obtained can yield suboptimal out-of-sample performance if the adopted model is biased. In this paper, we take into account both nurse demand and absenteeism uncertainty in a multi-unit and multi-pool setting. To address the challenges on computational scalability and out-of-sample performance, we propose an alternative nonparametric model based on DRO. In particular, this model allows dependence or independence between the absence rate and the staffing level. Moreover, our model can be solved to global optimality by a separation algorithm and, in several important special cases, by solving a single mixed-integer linear program (MILP).

[1] More precisely, the model in [14] allows to re-assign nurses from one unit to any other unit. In the context of this paper, that is equivalent to having a single float pool that serves all the units and assigning all nurses to this pool.

DRO models have received increasing attention in the recent literature. In particular, as in this paper, DRO has been applied to model two-stage stochastic optimization problems (see, e.g., [4, 5, 17]). In general, the two-stage DRO models are computationally prohibitive. For example, suppose that the second-stage formulation is linear and continuous with right-hand side uncertainty. Then, even with *fixed* first-stage decision variables, [4] showed that evaluating the objective function of the DRO model is NP-hard. To mitigate the computational challenge, [17, 24] recast the two-stage DRO model as a copositive program, which admits semidefinite programming approximations. In addition, [1, 5] applied linear decision rules (LDRs) to obtain conservative and tractable approximations. In contrast to these work, our second-stage formulation involves integer variables to model the pre-shift staffing. Besides undermining the convexity of our formulation, this prevents us from applying the LDRs because fractional staffing levels are not implementable. To the best of our knowledge, there are only two existing work [27, 32] on DRO with endogenous uncertainty. Specifically, [27] derived equivalent reformulations of the endogenous DRO model under various ambiguity sets, and [32] applied an endogenous DRO model on the machine scheduling problem. In this paper, we study a two-stage endogenous DRO model for nurse staffing and derive tractable reformulations under several practical float pool structures. We summarize our main contributions as follows:

1. We propose the first DRO approach for nurse staffing, considering both exogenous nurse demand and endogenous nurse absenteeism. The proposed two-stage endogenous DRO model considers multiple units, multiple float pools, and both long-term and pre-shift nurse staffing. For arbitrary pool structures, we derive a min-max reformulation of the model and a separation algorithm that solves this model to global optimality.

2. For multiple pool structures that often arise in practice, including one pool, disjoint pools, and chained pools, we provide a monolithic MILP reformulation of our DRO model by deriving strong valid inequalities. The binary variables of this MILP reformulation arise from the nurse staffing decisions only. That is, under these practical pool structures, the computational burden of our DRO approach is de facto the same as that of the deterministic nurse staffing.

3. Building upon the DRO model, we further study how to design sparse and effective disjoint pools. To this end, we proactively optimize the nurse pool structure to minimize the total number of cross-training, while providing a guarantee on the staffing cost.

4. We conduct extensive case studies based on the data and insights from our collaborating hospital. The results demonstrate the value of modeling nurse absenteeism and the computational efficacy of our DRO approach. In addition, we provide managerial insights on how to design sparse and effective pools.

The remainder of the paper is organized as follows. In Section 2, we describe the two-stage DRO model with endogenous nurse absenteeism. In Section 3, we derive a solution approach for this model under arbitrary pool structures. In Section 4, we derive strong valid inequalities and tractable reformulations under special pool structures. We extend the DRO model for optimal pool design in Section 5, conduct case studies in Section 6, and conclude in Section 7. To ease the exposition, we relegate all proofs to the appendices.

Notation: We use $\sim$ to indicate random variables and $\wedge$ to indicate realizations of the random variables. For example, $\tilde{d}$ represents a random variable and $\hat{d}^1, \ldots, \hat{d}^N$ represent N realizations of $\tilde{d}$. For $a, b \in \mathbb{Z}$, we define $[a] := \{1, 2, \ldots, a\}$ and $[a, b]_{\mathbb{Z}} := \{n \in \mathbb{Z} : a \leq n \leq b\}$. For $x \in \mathbb{R}$, we define $[x]_+ = \max\{x, 0\}$. For set S, we define its indicator function $\mathbb{1}_S$ such that $\mathbb{1}_S(s) = 1$ if $s \in S$ and $\mathbb{1}_S(s) = 0$ if $s \notin S$, and denote its convex hull by $\text{conv}(S)$.

2 Distributionally Robust Nurse Staffing

We consider a group of J hospital units, each facing a random demand of nurses denoted by $\tilde{d}_j$ for all $j \in [J]$. To enhance the operational flexibility, the manager forms I nurse float pools. For all $i \in [I]$, pool i is associated with a set P_i of units and each nurse assigned to this pool is capable of working in any unit $j \in P_i$. Due to random absenteeism, if we staff unit j with w_j nurses (termed unit nurses), then there will be a random number $\tilde{w}_j$ of nurses showing up for work, where $\tilde{w}_j \in [0, w_j]_{\mathbb{Z}}$. Likewise, $\tilde{y}_i$ nurses show up if we staff pool i with y_i nurses, where $\tilde{y}_i \in [0, y_i]_{\mathbb{Z}}$. After the uncertain parameters $\tilde{d}_j$, $\tilde{w}_j$, and $\tilde{y}_i$ are realized, the nurses showing up in pool i can be re-assigned to any units in P_i to make up the nurse shortage, if any. After the re-assignment, any remaining shortage will be covered by hiring temporary nurses in order to meet the NPR requirement. Mathematically, for given $\tilde{w} := [\tilde{w}_1, \ldots, \tilde{w}_J]^\top$, $\tilde{y} := [\tilde{y}_1, \ldots, \tilde{y}_I]^\top$, and $\tilde{d} := [\tilde{d}_1, \ldots, \tilde{d}_J]^\top$, the total operational cost can be obtained from solving the following integer program:

$$V(\tilde{w}, \tilde{y}, \tilde{d}) = \min_{z,x,e} \sum_{j=1}^{J} \left(c^{\mathrm{x}} x_j - c^{\mathrm{e}} e_j\right) \tag{1a}$$

$$\text{s.t.} \sum_{i:\ j \in P_i} z_{ij} + x_j - e_j = \tilde{d}_j - \tilde{w}_j, \quad \forall j \in [J], \tag{1b}$$

$$\sum_{j \in P_i} z_{ij} \leq \tilde{y}_i, \quad \forall i \in [I], \tag{1c}$$

$$x_j, e_j \in \mathbb{Z}_+, \quad \forall j \in [J], \quad z_{ij} \in \mathbb{Z}_+, \quad \forall i \in [I],\ \forall j \in P_i, \tag{1d}$$

where variables z_{ij} represent the number of nurses re-assigned from pool i to unit j, variables x_j represent the number of temporary nurses hired in unit j, variables e_j represent the excessive number of nurses in unit j, parameter c^{x} represents the unit cost of hiring temporary nurses, and parameter c^{e} represents the unit benefit of having excessive nurses. We can set c^{e} to be zero when such benefit is not taken into account. In the above formulation, objective function (1a) minimizes the cost of hiring temporary nurses minus the benefit of having excessive nurses. Constraints (1b) describe three ways of satisfying the nurse demand in each unit: (i) assigning unit nurses, (ii) re-assigning pool nurses, and (iii) hiring temporary nurses. Constraints (1c) ensure that the number of nurses re-assigned from each pool does not exceed the number of nurses showing up in that pool. Constraints (1d) describe integrality restrictions.

In reality, it is often challenging to obtain an accurate estimate of the true probability distribution $\mathbb{P}_{\tilde{w},\tilde{y},\tilde{d}}$ of $(\tilde{w}, \tilde{y}, \tilde{d})$. For example, the historical data of the nurse demand (via patient census and NPRs) can typically be explained by multiple (drastically) different distributions. More importantly, because of the endogeneity of $\tilde{w}$ and $\tilde{y}$, $\mathbb{P}_{\tilde{w},\tilde{y},\tilde{d}}$ is in fact a *conditional distribution* depending on the nurse staffing levels. This further increases the difficulty of estimation. Using a biased estimate of $\mathbb{P}_{\tilde{w},\tilde{y},\tilde{d}}$ can yield post-decision disappointment. For example, if one simply ignores the endogeneity of $\tilde{w}$ and $\tilde{y}$ and employs their empirical distribution based on historical data, then the nurse staffing thus obtained may lead to disappointing out-of-sample performance. In this paper, we assume that $\mathbb{P}_{\tilde{w},\tilde{y},\tilde{d}}$ is ambiguous and it belongs to the following moment ambiguity set:

$$\mathcal{D} \;=\; \Big\{\mathbb{P} \in \mathcal{P}(\Xi) : \;\; \mathbb{E}_{\mathbb{P}}[\tilde{d}_j^q] = \mu_{jq}, \quad \forall j \in [J], \; \forall q \in [Q], \tag{2a}$$

$$\mathbb{E}_{\mathbb{P}}[\tilde{w}_j] = f_j(w_j), \quad \forall j \in [J], \tag{2b}$$

$$\mathbb{E}_{\mathbb{P}}[\tilde{y}_i] = g_i(y_i), \quad \forall i \in [I]\Big\}, \tag{2c}$$

where Ξ represents the support of $(\tilde{w}, \tilde{y}, \tilde{d})$ and $\mathcal{P}(\Xi)$ represents the set of probability distribution supported on Ξ. We consider a box support $\Xi := \Xi_{\tilde{w}} \times \Xi_{\tilde{y}} \times \Xi_{\tilde{d}}$, where $\Xi_{\tilde{w}} = \Pi_{j=1}^{J}[0, w_j]_{\mathbb{Z}}$, $\Xi_{\tilde{y}} = \Pi_{i=1}^{I}[0, y_i]_{\mathbb{Z}}$, $\Xi_{\tilde{d}} = \Pi_{j=1}^{J}[d_j^{\text{L}}, d_j^{\text{U}}]_{\mathbb{Z}}$, and d_j^{L} and d_j^{U} represent lower and upper bounds of the nurse demand in unit j. In addition, for $Q \in \mathbb{N}_+$, all $q \in [Q]$, and all $j \in [J]$, μ_{jq} represents the q^{th} moment of $\tilde{d}_j$. Furthermore, for all $j \in [J]$ and $i \in [I]$, $f_j : \mathbb{N}_+ \to \mathbb{R}_+$ and $g_i : \mathbb{N}_+ \to \mathbb{R}_+$ represent two functions such that $f_j(0) = g_i(0) = 0$. We note that these functions can model arbitrary dependence of $(\tilde{w}, \tilde{y})$ on the staffing levels, and the assumption $f_j(0) = g_i(0) = 0$ ensures that if we assign no nurses in a unit/pool then nobody will show up.

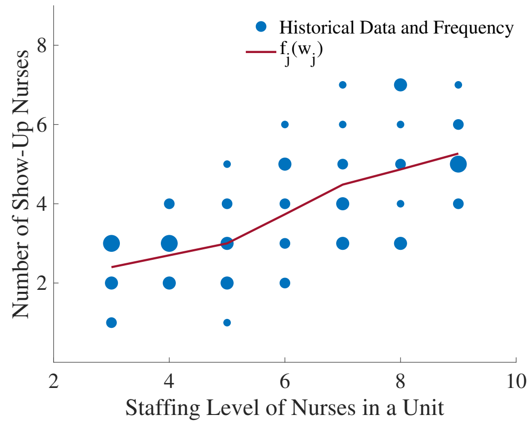


Figure 1: An example of segmented linear regression. Dots represent historical data samples and the size of dots indicate frequency.

The ambiguity set $\mathcal{D}$ can be conveniently calibrated. First, suppose that $\mathbb{P}_{\tilde{w},\tilde{y},\tilde{d}}$ is observed through nurse demand data $\{\hat{d}_j^1, \ldots, \hat{d}_j^N\}_{j=1}^J$ and attendance records $\{(w_j^1, \hat{w}_j^1), \ldots, (w_j^N, \hat{w}_j^N)\}_{j=1}^J$ and $\{(y_i^1, \hat{y}_j^1), \ldots, (y_i^N, \hat{y}_i^N)\}_{i=1}^I$ during the past N days, where, in each pair $(w_j^n, \hat{w}_j^n)$, w_j^n represents the staffing level of unit j in day n and $\hat{w}_j^n$ represents the corresponding number of nurses who actually showed up. Then, μ_{jq} can be obtained from empirical estimates (e.g., $\mu_{j1} = (1/N)\sum_{n=1}^N \hat{d}_j^n$, $\mu_{j2} = (1/N)\sum_{n=1}^N (\hat{d}_j^n)^2$, etc.), and f_j and g_i can be obtained by performing segmented linear regression on the attendance data, using the staffing levels $\{w_j^1, \ldots, w_j^N\}$ and $\{y_i^1, \ldots, y_i^N\}$ as breakpoints, respectively (see Figure 1 for an example). Second, if $\tilde{w}$ and $\tilde{y}$ are believed to follow certain parametric models, then we can follow such models to calibrate $\{f_j(w_j)\}_{j=1}^J$ and $\{g_i(y_i)\}_{i=1}^I$. For example, if $\tilde{w}_j$ is modeled as a Binomial random variable $B(w_j, 1-a(w_j))$ as in [16], where $a(w_j)$ represents the absence rate, i.e., the probability of any scheduled nurse in unit j being absent from work, then we have $f_j(w_j) = w_j(1-a(w_j))$.

We seek nurse staffing levels that minimize the expected total cost with regard to the worst-case probability distribution in $\mathcal{D}$, i.e., we consider the following two-stage DRO model:

$$\textbf{(DRNS)}: \quad \min_{w,y} \ \sum_{j=1}^J c^{\mathrm{w}} w_j + \sum_{i=1}^I c^{\mathrm{y}} y_i + \sup_{\mathbb{P}\in\mathcal{D}} \mathbb{E}_{\mathbb{P}}\left[V(\tilde{w},\tilde{y},\tilde{d})\right] \tag{3a}$$

$$\text{s.t.} \quad w_j^{\mathrm{L}} \le w_j \le w_j^{\mathrm{U}}, \quad \forall j \in [J], \tag{3b}$$

$$y_i^{\mathrm{L}} \le y_i \le y_i^{\mathrm{U}}, \quad \forall i \in [I], \tag{3c}$$

$$y, w \in R \cap \mathbb{Z}_+^{I+J}, \tag{3d}$$

where parameters c^{w} and c^{y} represent the unit cost of hiring unit and pool nurses, respectively, constraints (3b)–(3c) designate lower and upper bounds on staffing levels, and set R represents all remaining restrictions, which we assume can be represented via mixed-integer linear inequalities.

(DRNS) is computationally challenging because (i) $\mathcal{D}$ involves exponentially many probability distributions, all of which depend on the decision variables w_j and y_i and (ii) it is a two-stage DRO model with integer recourse variables. In the next two sections, we shall derive equivalent reformulations of (DRNS) that facilitate a separation algorithm, and identify practical pool structures that admit more tractable solution approaches.

3 Solution Approach: Arbitrary Pool Structure

In this section, we consider arbitrary pool structures, recast (DRNS) as a min-max formulation, and derive a separation algorithm that solves this model to global optimality.

We start by noticing that the integrality restrictions (1d) in the second-stage formulation of (DRNS) can be relaxed without loss of generality.

Lemma 1 *For any given $(\tilde{w}, \tilde{y}, \tilde{d}) \in \Xi$, the value of $V(\tilde{w}, \tilde{y}, \tilde{d})$ remains unchanged if constraints* (1d) *are replaced by non-negativity restrictions, i.e., $x_j, e_j \geq 0, \forall j \in [J]$ and $z_{ij} \geq 0, \forall i \in [I], \ \forall j \in P_i$.*

Thanks to Lemma 1, we are able to rewrite $V(\tilde{w}, \tilde{y}, \tilde{d})$ as the following dual formulation:

$$V(\tilde{w}, \tilde{y}, \tilde{d}) \ = \ \max_{\alpha, \beta} \ \sum_{j=1}^{J} (\tilde{d}_j - \tilde{w}_j)\alpha_j + \sum_{i=1}^{I} \tilde{y}_i \beta_i \tag{4a}$$

$$\text{s.t.} \ \ \beta_i + \alpha_j \leq 0, \quad \forall i \in [I], \ \forall j \in P_i, \tag{4b}$$

$$c^{\mathrm{e}} \leq \alpha_j \leq c^{\mathrm{x}}, \quad \forall j \in [J], \tag{4c}$$

where dual variables α_j and β_i are associated with primal constraints (1b) and (1c), respectively, and dual constraints (4b) and (4c) are associated with primal variables z_{ij} and (x_j, e_j), respectively. We let Λ denote the dual feasible region for variables (α, β), i.e., $\Lambda := \{(\alpha, \beta) : \text{(4b)–(4c)}\}$. Strong duality between formulations (1a)–(1d) and (4a)–(4c) hold valid because (1a)–(1d) has a finite optimal value.

We are now ready to recast (DRNS) as a min-max formulation. To this end, we consider $\mathbb{P}$ as a decision variable and take the dual of the worst-case expectation in (3a). For strong duality, we make the following technical assumption on the ambiguity set $\mathcal{D}$.

Assumption 1 *For any given $w := [w_1, \ldots, w_J]^\top$ and $y := [y_1, \ldots, y_I]^\top$ that are feasible to (DRNS), $\mathcal{D}$ is non-empty.*

Assumption 1 is mild. For example, it holds valid whenever the moments of demands $\{\mu_{jq} : j \in [J], q \in [Q]\}$ are obtained from empirical estimates and the decision-depend moments $\{g_i(y_i), f_j(w_j) : i \in [I], j \in [J]\}$ lie in the convex hull of their support, i.e., $f_j(w_j) \in [0, w_j]$ and $g_i(y_i) \in [0, y_i]$. In Appendix B, we present an approach to verify Assumption 1 by solving J linear programs. The reformulation is summarized in the following proposition.

Proposition 1 *Under Assumption 1, the (DRNS) model* (3) *yields the same optimal value and the same set of optimal solutions as the following min-max optimization problem:*

$$\min_{\substack{w,y\\ \gamma,\lambda,\rho}} \max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta) + \sum_{i=1}^{I}(c^y y_i + g_i(y_i)\lambda_i) + \sum_{j=1}^{J}\left[c^w w_j + \sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + f_j(w_j)\gamma_j\right] \tag{5a}$$

$$\text{s.t. } \text{(3b)–(3d)}, \tag{5b}$$

where

$$F(\alpha,\beta) := \sum_{j=1}^{J}\Big[(-\alpha_j-\gamma_j)w_j\Big]_+ + \sum_{i=1}^{I}\Big[(\beta_i-\lambda_i)y_i\Big]_+ + \sum_{j=1}^{J}\sup_{\tilde{d}_j\in[d_j^L,d_j^U]_{\mathbb{Z}}}\Big\{\alpha_j\tilde{d}_j - \sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q\Big\}. \tag{5c}$$

In the min-max reformulation (5a)–(5b), the additional variables γ, λ, ρ are generated in the process of taking dual. In addition, function $F(\alpha,\beta)$ is jointly convex in (α,β) because, as presented in (5c), $F(\alpha,\beta)$ is the pointwise maximum of functions affine in (α,β). This min-max reformulation is not directly computable because (i) for fixed $(w,y,\gamma,\lambda,\rho)$, evaluating the objective function (5a) needs to solve a *convex maximization* problem $\max_{(\alpha,\beta)\in\Lambda}F(\alpha,\beta)$, which is in general NP-hard, and (ii) the formulation includes nonlinear and non-convex terms $g_i(y_i)\lambda_i$ and $f_j(w_j)\gamma_j$. We shall address these two challenges before presenting a separation algorithm for solving (DRNS).

First, we analyze the convex maximization problem $\max_{(\alpha,\beta)\in\Lambda}F(\alpha,\beta)$ and derive the following optimality conditions.

Lemma 2 *For fixed $(w,y,\gamma,\lambda,\rho)$, there exists an optimal solution $(\bar{\alpha},\bar{\beta})$ to problem $\max_{(\alpha,\beta)\in\Lambda}F(\alpha,\beta)$ such that (a) $\bar{\alpha}_j\in\{c^e,c^x\}$ for all $j\in[J]$ and (b) $\bar{\beta}_i = -\max\{\bar{\alpha}_j : j\in P_i\}$ for all $i\in[I]$.*

Lemma 2 enables us to avoid enumerating the infinite number of elements in Λ and focus only on a finite set of $(\bar{\alpha},\bar{\beta})$ values. In addition, we introduce binary variables to encode the special structure identified in the optimality conditions. Specifically, for all $j\in[J]$, we define binary variables t_j such that $t_j=1$ if $\bar{\alpha}_j=c^x$ and $t_j=0$ if $\bar{\alpha}_j=c^e$; and for all $i\in[I]$ and $j\in P_i$, binary variables $s_{ij}=1$ if j is the largest index in P_i such that $t_j=1$ (i.e., $\bar{\alpha}_j=-c^x$ and $\bar{\alpha}_\ell=-c^e$ for all $\ell\in P_i$ and $\ell\geq j+1$) and $s_{ij}=0$ otherwise. Variables (t,s) need to satisfy the following constraints to make the encoding well-defined:

$$\sum_{j\in P_i}s_{ij}\leq 1,\quad \forall i\in[I], \tag{6a}$$

$$s_{ij}\leq t_j,\quad \forall i\in[I],\ \forall j\in P_i, \tag{6b}$$

$$t_j+s_{i\ell}\leq 1,\quad \forall i\in[I],\ \forall j,\ell\in P_i \text{ and } j>\ell, \tag{6c}$$

$$t_j\leq\sum_{\ell\in P_i}s_{i\ell},\quad \forall i\in[I],\ \forall j\in P_i, \tag{6d}$$

$$t_j\in\mathbb{B},\quad \forall j\in[J],\quad s_{ij}\in\mathbb{B},\quad \forall i\in[I],\ \forall j\in P_i, \tag{6e}$$

where constraints (6a) describe that, for all $i \in [I]$, $s_{ij} = 1$ holds for at most one $j \in P_i$, constraints (6b) designate that if $s_{ij} = 1$ then $t_j = 1$ because of the definition of s_{ij}, constraints (6c) describe that, for any two indices $j, \ell \in P_i$ with $j > \ell$, if $s_{i\ell} = 1$ then $t_j = 0$ because ℓ is the largest index such that $t_\ell = 1$, and constraints (6d) ensure that $\bar{\alpha}_j = c^{\mathrm{e}}$ for all $j \in P_i$ if all $s_{ij} = 0$. It follows that $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$ can be recast as an integer linear program presented in the following theorem. For the ease of exposition, we introduce dependent variables $r_j \equiv 1 - t_j$ and $p_i \equiv 1 - \sum_{j \in P_i} s_{ij}$.

Theorem 1 *For fixed $(w, y, \gamma, \lambda, \rho)$, problem $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$ yields the same optimal value as the following integer linear program:*

$$\max_{t,s,r,p} \quad \sum_{j=1}^{J} \left(c_j^t t_j + c_j^r r_j\right) + \sum_{i=1}^{I} \Big(c_i^p p_i + \sum_{j \in P_i} c_i^s s_{ij}\Big) \tag{7a}$$

$$\text{s.t.} \quad (t, s, r, p) \in \mathcal{H} \ := \ \Big\{ \text{(6a)–(6e)}, \tag{7b}$$

$$t_j + r_j = 1, \quad \forall j \in [J], \tag{7c}$$

$$\sum_{j \in P_i} s_{ij} + p_i = 1, \quad \forall i \in [I] \Big\}, \tag{7d}$$

where $c_j^t := \big[(-c^x - \gamma_j) w_j\big]_+ + \sup_{\tilde{d}_j \in [d_j^L, d_j^U]_{\mathbb{Z}}} \{c^x \tilde{d}_j - \sum_{q=1}^{Q} \rho_{jq} \tilde{d}_j^q\}$, $c_j^r := \big[(-c^e - \gamma_j) w_j\big]_+ + \sup_{\tilde{d}_j \in [d_j^L, d_j^U]_{\mathbb{Z}}} \{c^e \tilde{d}_j - \sum_{q=1}^{Q} \rho_{jq} \tilde{d}_j^q\}$, $c_i^p := \big[(-c^e - \lambda_i) y_i\big]_+$, *and* $c_i^s := \big[(-c^x - \lambda_i) y_i\big]_+$.

Second, we linearize the terms $f_j(w_j)\gamma_j$ and $g_i(y_i)\lambda_i$. For all $j \in [J]$, although $f_j(w_j)$ can be nonlinear and non-convex, thanks to the integrality of w_j, we can rewrite $f_j(w_j)$ as an affine function based on a binary expansion of w_j. Specifically, we introduce binary variables $\{u_{jk} : k \in [w_j^{\mathrm{U}} - w_j^{\mathrm{L}}]\}$ such that $w_j = w_j^{\mathrm{L}} + \sum_{k=1}^{w_j^{\mathrm{U}} - w_j^{\mathrm{L}}} u_{jk}$, where we interpret u_{jk} as whether we assign at least $w_j^{\mathrm{L}} + k$ nurses to unit j. That is, $u_{jk} = 1$ if $w_j \geq w_j^{\mathrm{L}} + k$ and $u_{jk} = 0$ otherwise. Then, defining $\Delta_{jk} := f_j(w_j^{\mathrm{L}} + k) - f_j(w_j^{\mathrm{L}} + k - 1)$ for all $k \in [w_j^{\mathrm{U}} - w_j^{\mathrm{L}}]$, we have

$$\begin{aligned} f_j(w_j) &= f_j(w_j^{\mathrm{L}}) + \sum_{k=1}^{w_j - w_j^{\mathrm{L}}} \left[f_j(w_j^{\mathrm{L}} + k) - f_j(w_j^{\mathrm{L}} + k - 1)\right] \\ &= f_j(w_j^{\mathrm{L}}) + \sum_{k=1}^{w_j^{\mathrm{U}} - w_j^{\mathrm{L}}} \left[f_j(w_j^{\mathrm{L}} + k) - f_j(w_j^{\mathrm{L}} + k - 1)\right] \mathbb{1}_{[w_j^{\mathrm{L}} + k, w_j^{\mathrm{U}}]}(w_j) \\ &= f_j(w_j^{\mathrm{L}}) + \sum_{k=1}^{w_j^{\mathrm{U}} - w_j^{\mathrm{L}}} \Delta_{jk} u_{jk}. \end{aligned}$$

It follows that $f_j(w_j)\gamma_j = f_j(w_j^{\mathrm{L}})\gamma_j + \sum_{k=1}^{w_j^{\mathrm{U}} - w_j^{\mathrm{L}}} \Delta_{jk} u_{jk} \gamma_j$. We can linearize the bilinear terms $u_{jk}\gamma_j$ by defining continuous variables $\varphi_{jk} := u_{jk}\gamma_j$ and incorporating the following standard McCormick inequalities (see [30]):

$$\gamma_j - M(1 - u_{jk}) \leq \varphi_{jk} \leq M u_{jk}, \quad \forall j \in [J], \ \forall k \in [w_j^{\mathrm{U}} - w_j^{\mathrm{L}}], \tag{8a}$$

$$-Mu_{jk} \le \varphi_{jk} \le \gamma_j + M(1-u_{jk}), \quad \forall j \in [J],\ \forall k \in [w_j^{\mathrm{U}} - w_j^{\mathrm{L}}], \tag{8b}$$

where M represents a sufficiently large positive constant. Likewise, for all $i \in [I]$, we rewrite $g_i(y_i)\lambda_i$ as $g_i(y_i^{\mathrm{L}})\lambda_i + \sum_{\ell=1}^{y_i^{\mathrm{U}}-y_i^{\mathrm{L}}} \delta_{i\ell} v_{i\ell}\lambda_i$ by using constants $\delta_{i\ell} := g_i(y_i^{\mathrm{L}}+\ell) - g_i(y_i^{\mathrm{L}}+\ell-1)$ for all $\ell \in [y_i^{\mathrm{U}} - y_i^{\mathrm{L}}]$ and binary variables $\{v_{i\ell} : \ell \in [y_i^{\mathrm{U}} - y_i^{\mathrm{L}}]\}$, where $v_{i\ell} = 1$ if $y_i \ge y_i^{\mathrm{L}} + \ell$ and $v_{i\ell} = 0$ otherwise. We linearize the bilinear terms $v_{i\ell}\lambda_i$ by continuous variables $\nu_{i\ell} := v_{i\ell}\lambda_i$ and the McCormick inequalities

$$\lambda_i - M(1-v_{i\ell}) \le \nu_{i\ell} \le Mv_{i\ell}, \quad \forall i \in [I],\ \forall \ell \in [y_i^{\mathrm{U}} - y_i^{\mathrm{L}}], \tag{8c}$$

$$-Mv_{i\ell} \le \nu_{i\ell} \le \lambda_i + M(1-v_{i\ell}), \quad \forall i \in [I],\ \forall \ell \in [y_i^{\mathrm{U}} - y_i^{\mathrm{L}}]. \tag{8d}$$

In computation, a large big-M coefficient M can significantly slow down the solution of (DRNS). Theoretically, for the correctness of the linearization (8a)–(8d), M needs to be larger than $|\gamma_j|$ and $|\lambda_i|$ for all $j \in [J]$ and $i \in [I]$, respectively. The following proposition derives uniform lower and upper bounds of γ_j and λ_i, leading to a small value of M.

Proposition 2 *For fixed w and y, there exists an optimal solution $(\gamma^*, \lambda^*, \rho^*)$ to formulation* (5a)–(5b) *such that $\gamma_j^* \in [-c^x, 0]$ for all $j \in [J]$ and $\lambda_i^* \in [-c^x, 0]$ for all $i \in [I]$.*

Proposition 2 indicates that (i) we can set $M := c^{\mathrm{x}}$ in the McCormick inequalities (8a)–(8d) without loss of optimality and (ii) as all γ_j and λ_i are non-positive at optimality, we can replace McCormick inequalities (8a) and (8c) as $\gamma_j \le \varphi_{jk} \le 0$ and $\lambda_i \le \nu_{i\ell} \le 0$ respectively, both of which are now big-M-free. In addition, we incorporate the following constraints to break the symmetry among binary variables:

$$u_{jk} \ge u_{j(k+1)}, \quad \forall j \in [J],\ \forall k \in [w_j^{\mathrm{U}} - w_j^{\mathrm{L}} - 1], \tag{8e}$$

$$v_{i\ell} \ge v_{i(\ell+1)}, \quad \forall i \in [I],\ \forall \ell \in [y_i^{\mathrm{U}} - y_i^{\mathrm{L}} - 1]. \tag{8f}$$

The above analysis recasts (DRNS) into a mixed-integer program, which is summarized in the following theorem without proof.

Theorem 2 *Under Assumption 1, the (DRNS) model* (3) *yields the same optimal value as the following mixed-integer program:*

$$\begin{aligned}
\min_{\substack{u,v,\varphi,\nu\\ \gamma,\lambda,\rho,\theta}} \quad & \theta + \sum_{i=1}^{I}\Big(c^y y_i^L + g_i(y_i^L)\lambda_i + \sum_{\ell=1}^{y_i^U - y_i^L}\big(\delta_{i\ell}\nu_{i\ell} + c^y v_{i\ell}\big)\Big) \\
& + \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + c^w w_j^L + f_j(w_j^L)\gamma_j + \sum_{k=1}^{w_j^U - w_j^L}\big(c^w u_{jk} + \Delta_{jk}\varphi_{jk}\big)\right] && (9a)\\
s.t. \quad & (8a)\text{–}(8f), && (9b)\\
& \theta \ge \sum_{j=1}^{J}\big(c_j^t t_j + c_j^r r_j\big) + \sum_{i=1}^{I}\Big(c_i^p p_i + \sum_{j\in P_i} c_i^s s_i\Big), \quad \forall (t,s,r,p) \in \mathcal{H}, && (9c)
\end{aligned}$$

$$u_{jk} \in \mathbb{B}, \quad \forall j \in [J],\ \forall k \in [w_j^U - w_j^L], \quad v_{i\ell} \in \mathbb{B}, \quad \forall i \in [I],\ \forall \ell \in [y_i^U - y_i^L], \tag{9d}$$

where set $\mathcal{H}$ *is defined in* (7b)–(7d) *and coefficients* c_j^t, c_i^s, c_j^r, *and* c_i^p *are represented through*

$$c_j^t = \sup_{\tilde{d}_j \in [d_j^L, d_j^U]_{\mathbb{Z}}} \left\{ c^x \tilde{d}_j - \sum_{q=1}^{Q} \rho_{jq} \tilde{d}_j^q \right\}, \tag{9e}$$

$$c_i^s = 0, \tag{9f}$$

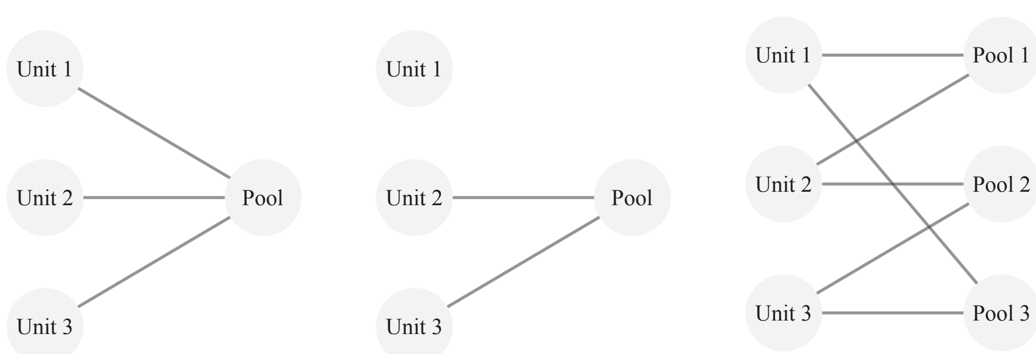
$$c_j^r = \left[(-c^e - \gamma_j) w_j^L - \sum_{k=1}^{w_j^U - w_j^L} (\varphi_{jk} + c^e u_{jk}) \right]_+ + \sup_{\tilde{d}_j \in [d_j^L, d_j^U]_{\mathbb{Z}}} \left\{ c^e \tilde{d}_j - \sum_{q=1}^{Q} \rho_{jq} \tilde{d}_j^q \right\}, \tag{9g}$$

$$\text{and} \quad c_i^p = \left[(-c^e - \lambda_i) y_i^L - \sum_{\ell=1}^{y_i^U - y_i^L} (\nu_{i\ell} + c^e v_{i\ell}) \right]_+. \tag{9h}$$

The reformulation (9a)–(9d) facilitates the separation algorithm (see, e.g., [31]), also known as delayed constraint generation. We notice that (9c) involve 2^J many constraints, making it computationally prohibitive to solve (9a)–(9d) in one shot. Instead, the separation algorithm incorporates constraints (9c) on-the-fly. Specifically, this algorithm first solves a relaxation of the reformulation by overlooking constraints (9c). Then, we check if the optimal solution thus obtained violates any of (9c). If yes, then we add one violated constraint back into the relaxation and re-solve. We call this added constraint a "cut" and note that each cut describes a convex feasible region. This procedure is repeat until an optimal solution is found to satisfy all of constraints (9c). We present the pseudo code in Algorithm 1.

We close this section by confirming the correctness of Algorithm 1.

Theorem 3 *Algorithm 1 finds a globally optimal solution to the (DRNS) model* (3) *in a finite number of iterations.*



(a) Structure [1] (b) Structure [D] (c) Structure [C]

Figure 2: Examples of practical pool structures

Algorithm 1 A Separation Algorithm for Solving the (DRNS) model (3)

1: Initialization: Set the set of cuts $\mathcal{H}_{\text{sep}} = \emptyset$.
2: Solve the master problem

$$
\begin{aligned}
(\mathbf{MP}): \quad \min_{\substack{u,v,\varphi,\nu \\ \gamma,\lambda,\rho}} \quad & \theta + \sum_{i=1}^{I} \Big(c^{\text{y}} y_i^{\text{L}} + g_i(y_i^{\text{L}})\lambda_i + \sum_{\ell=1}^{y_i^{\text{U}} - y_i^{\text{L}}} \big(\delta_{i\ell}\nu_{i\ell} + c^{\text{y}} v_{i\ell} \big) \Big) \\
& + \sum_{j=1}^{J} \left[\sum_{q=1}^{Q} \mu_{jq}\rho_{jq} + c^{\text{w}} w_j^{\text{L}} + f_j(w_j^{\text{L}})\gamma_j + \sum_{k=1}^{w_j^{\text{U}} - w_j^{\text{L}}} \big(c^{\text{w}} u_{jk} + \Delta_{jk}\varphi_{jk} \big) \right] \\
\text{s.t.} \quad & \text{(9b), (9d)}, \\
& \theta \geq \sum_{j=1}^{J} \big(c_j^{\text{t}} t_j + c_j^{\text{r}} r_j \big) + \sum_{i=1}^{I} \big(c_i^{\text{p}} p_i + \sum_{j \in P_i} c_i^{\text{s}} s_{ij} \big), \quad \forall (t,s,r,p) \in \mathcal{H}_{\text{sep}},
\end{aligned}
$$

and record an optimal solution $(u^*, v^*, \varphi^*, \nu^*, \gamma^*, \lambda^*, \rho^*, \theta^*)$.
3: Compute $c_j^{\text{t}*}$, $c_i^{\text{s}*}$, $c_j^{\text{r}*}$, and $c_i^{\text{p}*}$ based on (9e)–(9h) and the values of $(u^*, v^*, \varphi^*, \nu^*, \gamma^*, \lambda^*, \rho^*)$.
4: Solve the integer linear program (7a)–(7d) using objective coefficients $c_j^{\text{t}*}$, $c_i^{\text{s}*}$, $c_j^{\text{r}*}$, and $c_i^{\text{p}*}$. Record an optimal solution (t^*, s^*, r^*, p^*).
5: **if** $\theta^* \geq \sum_{j=1}^{J} (c_j^{\text{t}*} t_j^* + c_j^{\text{r}*} r_j^*) + \sum_{i=1}^{I} (c_i^{\text{p}*} p_i^* + \sum_{j \in P_i} c_i^{\text{s}*} s_{ij}^*)$ **then**
6: Stop and return (u^*, v^*) as an optimal solution to (DRNS).
7: **else**
8: Add a cut in the form of (9c) into (**MP**) by setting $\mathcal{H}_{\text{sep}} \leftarrow \mathcal{H}_{\text{sep}} \cup \{(t^*, s^*, r^*, p^*)\}$. Go To Step 2.
9: **end if**

4 Tractable Cases: Practical Pool Structures

In this section, we consider the following three nurse pool structures that often arise in reality.

Structure [1] (One Pool) $I = 1$, *i.e., there is one single nurse pool shared among all units* (see Figure 2a for an example).

Structure [D] (Disjoint Pools) *All nurse pools are disjoint, i.e., for all* $i_1, i_2 \in [I]$ *and* $i_1 \neq i_2$, *it holds that* $P_{i_1} \cap P_{i_2} = \emptyset$ (see Figure 2b for an example).

Structure [C] (Chained Pools) *The nurse pools form a long chain, i.e., there are* $I = J$ *pools with* $P_i = \{i, i+1\}$ *for all* $i \in [I-1]$ *and* $P_I = \{I, 1\}$ (see Figure 2c for an example).

Structure [1] can be utilized when all units have similar functionalities and so they can all share one nurse pool. Accordingly, every nurse assigned to this pool should be cross-trained for all units so that he/she is able to undertake the tasks in them. Structure [D] is less demanding than one pool, as each pool covers only a subset of units which, e.g., have distinct functionalities. Accordingly, the amount of cross-training under this structure significantly decreases from that under one pool.

Structure [C] has been applied in the production systems to increase the operational flexibility (see, e.g., [20, 42, 11, 10]). Under this structure, every unit is covered by two nurse pools. Accordingly, every pool nurse needs to be cross-trained for only two units. All three structures have been considered and compared in a nurse staffing context (see, e.g., [19]). Under these practical pool structures, we derive tractable reformulations of the (DRNS) model (3). Our derivation leads to monolithic MILP reformulations that facilitate off-the-shelf software like GUROBI.

4.1 One Pool

We derive a valid inequality to strengthen feasible region $\mathcal{H}$ of the integer program (7a)–(7d).

Lemma 3 *Under any nurse pool structure, the following inequalities hold valid for all* $(t, s, r, p) \in \mathcal{H}$*:*

$$t_j \leq \sum_{\ell \in P_i : \ell \geq j} s_{i\ell}, \quad \forall i \in [I], \ \forall j \in P_i. \tag{10}$$

Under Structure [1], we show that inequalities (10), in conjunction with the existing constraints (7b)–(7d), are sufficient to describe the convex hull of $\mathcal{H}$. Better still, this yields a closed-form solution to the convex maximization problem $\max_{(\alpha,\beta)} F(\alpha, \beta)$.

Theorem 4 *Under Structure [1], it holds that conv*$(\mathcal{H}) = \overline{\mathcal{H}}$*, where*

$$\overline{\mathcal{H}} \ = \ \big\{(t, s, r, p) \geq 0 : \ \text{(6a)–(6b), (7c)–(7d), (10)}\big\}.$$

In addition, for fixed $(u, v, \gamma, \lambda, \rho)$*, problem* $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$ *yields the same optimal value as the linear program* $\max_{t,s,r,p}\big\{\sum_{j=1}^{J}\big(c^t_j t_j + c^r_j r_j\big) + c^p_1 p_1 : \ (t, s, r, p) \in \overline{\mathcal{H}}\big\}$ *and*

$$\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta) \ = \ \max\left\{c^p_1 + \sum_{\ell=1}^{J} c^r_\ell, \ \max_{j\in[J]}\left\{c^t_j + \sum_{\ell=1}^{j-1}\max\{c^t_\ell, c^r_\ell\} + \sum_{\ell=j+1}^{J} c^r_\ell\right\}\right\},$$

where c^t_j*,* c^r_j*, and* c^p_1 *are computed by* (9e)–(9h)*.*

Theorem 4 enables us to reduce the 2^J many constraints (9c) in the reformulation of (DRNS) to $(J+1)$ many, thanks to the closed-form solution of $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$. This leads to the following monolithic MILP reformulation of (DRNS).

Proposition 3 *Under Assumption 1 and Structure [1], the (DRNS) model* (3) *yields the same optimal objective value as the following MILP:*

$$Z^\star_{[1]} := \min \ \ \theta + c^y y^L_1 + g_1(y^L_1)\lambda_1 + \sum_{\ell=1}^{y^U_1 - y^L_1}\big(\delta_{1\ell}\nu_{1\ell} + c^y v_{1\ell}\big)$$

$$+ \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + c^w w^L_j + f_j(w^L_j)\gamma_j + \sum_{k=1}^{w^U_j - w^L_j}\big(c^w u_{jk} + \Delta_{jk}\varphi_{jk}\big)\right]$$

$$
\begin{aligned}
\textit{s.t.}\quad & \text{(8a)–(8f)},\ \text{(9d)},\\
& \theta \geq \phi_1 + \sum_{\ell=1}^{J} (\zeta_\ell + \eta^e_\ell), \quad \theta \geq \eta^x_j + \sum_{\ell}^{j-1} \chi_\ell + \sum_{\ell=j+1}^{J} (\zeta_\ell + \eta^e_\ell), \quad \forall j \in [J],\\
& \left.\begin{aligned}
& \chi_j \geq \zeta_j + \eta^e_j, \quad \chi_j \geq \eta^x_j\\
& \zeta_j \geq (-c^e - \gamma_j) w^L_j - \textstyle\sum_{k=1}^{w^U_j - w^L_j} (\varphi_{jk} + c^e u_{jk}), \quad \zeta_j \geq 0\\
& \eta^x_j \geq c^x \tilde{d}_j - \textstyle\sum_{q=1}^{Q} \tilde{d}^q_j \rho_{jq}, \quad \eta^e_j \geq c^e \tilde{d}_j - \textstyle\sum_{q=1}^{Q} \tilde{d}^q_j \rho_{jq}, \quad \forall \tilde{d}_j \in [d^L_j, d^U_j]_{\mathbb{Z}}
\end{aligned}\right\} \quad \forall j \in [J], \qquad (11)\\
& \phi_1 \geq (-c^e - \lambda_1) y^L_1 - \sum_{\ell=1}^{y^U_1 - y^L_1} (\nu_{1\ell} + c^e v_{1\ell}), \quad \phi_1 \geq 0.
\end{aligned}
$$

A special case of Structure [1] is when there are no nurse float pools. Mathematically, this is equivalent to assigning all units to one single pool with no pool nurses. We hence call it Structure [0] as there is zero pool nurse. Under this structure, $y^{\mathrm{L}}_1 = y^{\mathrm{U}}_1 = 0$ and accordingly $g_1(y^{\mathrm{L}}_1) = 0$. A MILP reformulation of (DRNS) under Structure [0] follows from Proposition 3:

$$
\begin{aligned}
Z^\star_{[0]} \ := \ \min\ & \theta + \sum_{j=1}^{J} \left[\sum_{q=1}^{Q} \mu_{jq} \rho_{jq} + c^{\mathrm{w}} w^{\mathrm{L}}_j + f_j(w^{\mathrm{L}}_j)\gamma_j + \sum_{k=1}^{w^{\mathrm{U}}_j - w^{\mathrm{L}}_j} \left(c^{\mathrm{w}} u_{jk} + \Delta_{jk} \varphi_{jk} \right) \right]\\
\text{s.t.}\ & \text{(8a)–(8b)},\ \text{(8e)},\ \text{(9d)},\ \text{(11)},\\
& \theta \geq \sum_{\ell=1}^{J} (\zeta_\ell + \eta^{\mathrm{e}}_\ell), \quad \theta \geq \eta^{\mathrm{x}}_j + \sum_{\ell}^{j-1} \chi_\ell + \sum_{\ell=j+1}^{J} (\zeta_\ell + \eta^{\mathrm{e}}_\ell), \quad \forall j \in [J].
\end{aligned}
$$

We notice that, whenever $y^{\mathrm{L}}_1 = 0$, any feasible nurse staffing levels under Structure [0] are also feasible to (DRNS) under Structure [1]. It then follows that $Z^\star_{[1]} \leq Z^\star_{[0]}$. In addition, as Structure [1] provides the most operational flexibility and Structure [0] has zero flexibility, we may interpret the difference $Z^\star_{[0]} - Z^\star_{[1]}$ as the (maximum) value of operational flexibility.

4.2 Disjoint Pools

Under Structure [D], we can once again obtain the convex hull of $\mathcal{H}$ by incorporating inequalities (10). Intuitively, as the nurse pools are disjoint, $\mathcal{H}$ becomes separable in index i, i.e., separable among the nurse pools and the units under each pool. Hence, conv($\mathcal{H}$) can be obtained by convexifying the projection of $\mathcal{H}$ in each pool and then taking their Cartesian product. It follows that, once again, the convex maximization problem $\max_{(\alpha,\beta)} F(\alpha, \beta)$ admits a closed-form solution and (DRNS) can be recast as a monolithic MILP. In particular, we reduce the exponentially many constraints (9c) in the reformulation of (DRNS) to $(I + J)$ many. We summarize these results in the following proposition.

Proposition 4 *Under Structure [D], it holds that conv($\mathcal{H}$) $= \overline{\mathcal{H}}$. In addition, for fixed* $(u, v, \gamma, \lambda, \rho)$, *it holds that*

$$\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta) \;=\; \sum_{i\in[I]} \max\Big\{c_i^p + \sum_{\ell\in P_i} c_\ell^r,\ \max_{j\in P_i}\Big\{c_j^t + \sum_{\ell\in P_i:\ell<j} \max\{c_\ell^t, c_\ell^r\} + \sum_{\ell\in P_i:\ell>j} c_\ell^r\Big\}\Big\},$$

where c_j^t, c_j^r, and c_i^p are computed by (9e)–(9h)*. Furthermore, under Assumption 1, the (DRNS) model* (3) *yields the same optimal objective value as the following MILP:*

$$\begin{aligned}
Z^\star_{[D]} \;:=\; \min \;\; & \sum_{i=1}^{I}\Big(\theta_i + c^y y_i^L + g_i(y_i^L)\lambda_i + \sum_{\ell=1}^{y_i^U-y_i^L} \big(\delta_{i\ell}\nu_{i\ell} + c^y v_{i\ell}\big)\Big) \\
& + \sum_{j=1}^{J}\Bigg[\sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + c^w w_j^L + f_j(w_j^L)\gamma_j + \sum_{k=1}^{w_j^U-w_j^L}\big(c^w u_{jk} + \Delta_{jk}\varphi_{jk}\big)\Bigg] \\
\text{s.t.} \;\; & \text{(8a)–(8f)},\ \text{(9d)},\ \text{(11)},\quad \theta_i \ge \phi_i + \sum_{\ell\in P_i}(\zeta_\ell + \eta_\ell^e),\quad \forall i\in[I], \\
& \theta_i \ge \eta_j^x + \sum_{\ell\in P_i:\ell<j}\chi_\ell + \sum_{\ell\in P_i:\ell>j}(\zeta_\ell + \eta_\ell^e),\quad \forall i\in[I],\ \forall j\in P_i, \\
& \phi_i \ge (-c^e - \lambda_i)y_i^L - \sum_{\ell=1}^{y_i^U-y_i^L}(\nu_{i\ell} + c^e v_{i\ell}),\quad \forall i\in[I], && \text{(12a)} \\
& \phi_i \ge 0,\quad \forall i\in[I],\quad \zeta_j \ge 0,\quad \forall j\in[J]. && \text{(12b)}
\end{aligned}$$

4.3 Chained Pools

Under Structure [C], the valid inequalities (10) can still be incorporated to strengthen and simplify the mixed-integer set $\mathcal{H}$. Specifically, as $P_i = \{i, i+1\}$ for all $i\in[I-1]$, inequalities (10) imply that $t_{i+1} \le s_{i(i+1)}$. But constraints (6b) designate $s_{i(i+1)} \le t_{i+1}$, implying that $t_{i+1} = s_{i(i+1)}$. Similarly, we obtain $s_{I1} = t_1$ and simplify $\mathcal{H}$ as follows:

$$\begin{aligned}
\mathcal{H} \;=\; \Big\{(t,s,r,p)\in\mathbb{B}^{4I} :\; & s_{ii} \le t_i \le s_{ii} + t_{\sigma(i)} \le 1,\quad \forall i\in[I], && (13)\\
& t_i + r_i = 1,\quad \forall i\in[I], \\
& p_i + s_{ii} + t_{\sigma(i)} = 1,\quad \forall i\in[I]\Big\},
\end{aligned}$$

where $\sigma(i) := i+1$ for all $i\in[I-1]$ and $\sigma(I) := 1$. Unfortunately, unlike under Structures [1] and [D], the strengthened $\mathcal{H}$ is no longer integral, i.e., $\text{conv}(\mathcal{H}) \ne \overline{\mathcal{H}}$. We demonstrate this fact in the following example.

Example 1 *Consider an example of 3 chained pools, i.e., $I = J = 3$, $P_1 = \{1,2\}$, $P_2 = \{2,3\}$, and $P_3 = \{3,1\}$. Incorporating valid inequalities* (10) *and relaxing integrality restrictions in $\mathcal{H}$ yields*

$$\begin{aligned}
\overline{\mathcal{H}} = \Big\{(t,s,r,p)\ge 0 :\; & s_{11} \le t_1 \le s_{11} + t_2 \le 1, && \text{(14a)}\\
& s_{22} \le t_2 \le s_{22} + t_3 \le 1, && \text{(14b)}
\end{aligned}$$

$$s_{33} \le t_3 \le s_{33} + t_1 \le 1, \qquad (14c)$$
$$t_1 + r_1 = t_2 + r_2 = t_3 + r_3 = 1,$$
$$\left. p_1 + s_{11} + t_2 = p_2 + s_{22} + t_3 = p_3 + s_{33} + t_1 = 1 \right\}.$$

We observe that polyhedron $\overline{\mathcal{H}}$ is 6-dimensional. Hence, replacing the first and last inequalities in constraints (14a)–(14c) *with equalities yields the following extreme point:*

$$(t_1, t_2, t_3, s_{11}, s_{22}, s_{33}, r_1, r_2, r_3, p_1, p_2, p_3) = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, 0, 0\right),$$

which is fractional. Therefore, $\overline{\mathcal{H}}$ is not integral and $\overline{\mathcal{H}} \neq conv(\mathcal{H})$.

Despite the loss of integrality, we adopt an alternative approach to recast the integer program (7a)–(7d), and hence the convex maximization problem $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$, as a linear program. We start by noticing that inequalities (13) allow us to represent variables s_{ii} as $s_{ii} = t_i(1 - t_{\sigma(i)})$ for all $i \in [I]$. In fact, (13) are exactly the McCormick inequalities that linearize this (nonlinear) representation. It follows that $p_i = 1 - s_{ii} - t_{\sigma(i)} = (1 - t_i)(1 - t_{\sigma(i)})$ for all $i \in [I]$. Plugging these representations into formulation (7a)–(7d) yields a reformulation of $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$ based on variables t only:

$$\begin{aligned}
\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta) &= \max_{t\in\mathbb{B}^I} \sum_{i=1}^{I} \Big[c_i^{\mathrm{t}} t_i + c_i^{\mathrm{r}}(1 - t_i) + c_i^{\mathrm{p}}(1 - t_i)(1 - t_{\sigma(i)})\Big] \\
&= \max_{t\in\mathbb{B}^I} \; c_1^{\mathrm{t}} t_1 + c_1^{\mathrm{r}}(1 - t_1) \\
&\quad + \sum_{i=2}^{I} \Big[c_i^{\mathrm{t}} + (c_i^{\mathrm{r}} - c_i^{\mathrm{t}})(1 - t_i) + c_{i-1}^{\mathrm{p}}(1 - t_{i-1})(1 - t_i)\Big] \\
&\quad + c_I^{\mathrm{p}}(1 - t_I)(1 - t_1). \qquad (15)
\end{aligned}$$

The reformulation (15) decomposes objective function based on index $i \in [I]$ and enables us to solve $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$ by a dynamic program (DP), i.e., we sequentially optimize $t_1, t_2, \ldots, t_I$. To this end, we define the state of the DP in stage 1 as $\widehat{t}_1 \in \mathbb{B}$ and the states in stage i as $(\widehat{t}_1, \widehat{t}_i) \in \mathbb{B}^2$ for all $i \in [2, I]_{\mathbb{Z}}$. In addition, we formulate the DP as $\max_{(\widehat{t}_1,\widehat{t}_I)\in\mathbb{B}^2} \{V_I(\widehat{t}_1, \widehat{t}_I) + c_I^{\mathrm{p}}(1 - \widehat{t}_I)(1 - \widehat{t}_1)\}$, where the value functions $V_i(\cdot)$ are recursively defined through

$$\begin{aligned}
V_1(\widehat{t}_1) &= c_1^{\mathrm{t}} \widehat{t}_1 + c_1^{\mathrm{r}}(1 - \widehat{t}_1), \\
\text{and} \quad V_i(\widehat{t}_1, \widehat{t}_i) &= \max_{\widehat{t}_{i-1}\in\mathbb{B}} \Big\{V_{i-1}(\widehat{t}_1, \widehat{t}_{i-1}) + c_i^{\mathrm{t}} + (c_i^{\mathrm{r}} - c_i^{\mathrm{t}})(1 - \widehat{t}_i) \\
&\qquad + c_{i-1}^{\mathrm{p}}(1 - \widehat{t}_{i-1})(1 - \widehat{t}_i)\Big\}, \quad \forall i \in [2, I]_{\mathbb{Z}}, \; \forall (\widehat{t}_1, \widehat{t}_i) \in \mathbb{B}^2.
\end{aligned}$$

For all $i \in [I]$, value function $V_i(\widehat{t}_1, \widehat{t}_i)$ represents the "cumulative reward" up to stage i, i.e., the terms in (15) that involve $t_1, \ldots, t_i$ only. We note that, as $\widehat{t}_1$ is involved in the final-stage reward, the DP stores the value of $\widehat{t}_1$ in the state throughout stages $2, \ldots, I$.

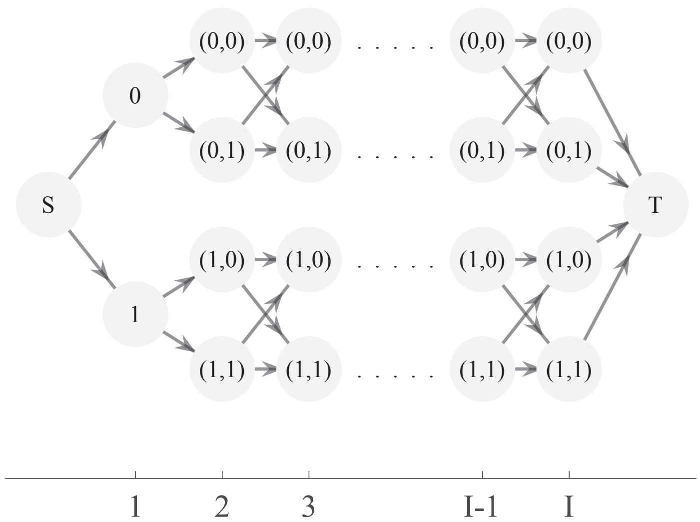


Figure 3: Longest-path problem on an acyclic direct network

We further interpret the DP as a longest-path problem on an acyclic directed network $(\mathcal{N}, \mathcal{A})$. Specifically, the set of nodes $\mathcal{N}$ consists of I layers, denoted by $\{\mathcal{N}_i\}_{i=1}^N$. For all $i \in [I]$, layer i consists of the states of the DP in stage i, i.e., $\mathcal{N}_1 = \{0, 1\}$, and $\mathcal{N}_i = \{(0,0),(0,1),(1,0),(1,1)\}$ for all $i \in [2, I]_{\mathbb{Z}}$. In addition, $\mathcal{A}$ consists of arcs that connect two nodes in neighboring layers, as long as the two nodes share a common $\widehat{t}_1$ value, i.e., $\mathcal{A} = \{[\widehat{t}_1, (\widehat{t}_1, \widehat{t}_2)] : \widehat{t}_1, \widehat{t}_2 \in \mathbb{B}\} \cup \{[(\widehat{t}_1, \widehat{t}_{i-1}), (\widehat{t}_1, \widehat{t}_i)] : \widehat{t}_{i-1}, \widehat{t}_i \in \mathbb{B}, \forall i \in [3, I]_{\mathbb{Z}}\}$. Finally, we incorporate into $\mathcal{N}$ a starting node S and a terminal node T, and into $\mathcal{A}$ arcs from S to all nodes in $\mathcal{N}_1$ and from all nodes in $\mathcal{N}_I$ to T. We depict $(\mathcal{N}, \mathcal{A})$ in Figure 3. Then, the DP is equivalent to the longest-path problem from S to T on $(\mathcal{N}, \mathcal{A})$. We formally state this result in the following theorem.

Theorem 5 *Define $\{c_{[m,n]} : [m,n] \in \mathcal{A}\}$, the length of the arcs in network $(\mathcal{N}, \mathcal{A})$, such that*

$$\begin{aligned}
c_{[\mathtt{S},\widehat{t}_1]} &= c_1^t \widehat{t}_1 + c_1^r (1 - \widehat{t}_1), \quad \forall \widehat{t}_1 \in \mathbb{B}, \\
c_{[(\widehat{t}_1,\widehat{t}_{i-1}),(\widehat{t}_1,\widehat{t}_i)]} &= c_i^t + (c_i^r - c_i^t)(1 - \widehat{t}_i) + c_{i-1}^p (1 - \widehat{t}_{i-1})(1 - \widehat{t}_i), \\
&\quad \forall \widehat{t}_{i-1}, \widehat{t}_i \in \mathbb{B}, \ \forall i \in [2, I]_{\mathbb{Z}}, \\
\text{and} \quad c_{[(\widehat{t}_1,\widehat{t}_I),\mathtt{T}]} &= c_I^p (1 - \widehat{t}_I)(1 - \widehat{t}_1), \quad \forall \widehat{t}_1, \widehat{t}_I \in \mathbb{B}.
\end{aligned}$$

Then, for fixed $(u, v, \gamma, \lambda, \rho)$, $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$ equals the length of the longest S-T path on $(\mathcal{N}, \mathcal{A})$, that is,

$$\begin{aligned}
\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta) \ = \ & \max_{x\in[0,1]^{\mathcal{A}}} \sum_{[m,n]\in\mathcal{A}} c_{[m,n]} x_{[m,n]} \\
& \quad \text{s.t.} \quad \sum_{n:[m,n]\in\mathcal{A}} x_{[m,n]} - \sum_{n:[n,m]\in\mathcal{A}} x_{[n,m]} = \begin{cases} 1, & \text{if } m = \mathtt{S} \\ 0, & \text{if } m \neq \mathtt{S}, \ \mathtt{T} \\ -1, & \text{if } m = \mathtt{T}, \end{cases} \quad \forall m \in \mathcal{N}.
\end{aligned}$$

We note that $(\mathcal{N}, \mathcal{A})$ is acyclic and it consists of $4I$ nodes and $8I-6$ arcs. Hence, the longest-path problem, as well as $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$, can be solved in time polynomial of the problem input. Accordingly, we are able to replace the exponentially many constraints (9c) in the reformulation of (DRNS) with $\mathcal{O}(I)$ many linear constraints. This yields the following monolithic MILP reformulation.

Proposition 5 *Under Structure [C] and Assumption 1, the (DRNS) model* (3) *yields the same optimal objective value as the following MILP:*

$$\begin{aligned}
\min \quad & \theta + \sum_{i=1}^{I}\Big(c^y y_i^L + g_i(y_i^L)\lambda_i + \sum_{\ell=1}^{y_i^U - y_i^L} \big(\delta_{i\ell}\nu_{i\ell} + c^y v_{i\ell}\big)\Big) \\
& + \sum_{j=1}^{I}\left[\sum_{q=1}^{Q} \mu_{jq}\rho_{jq} + c^w w_j^L + f_j(w_j^L)\gamma_j + \sum_{k=1}^{w_j^U - w_j^L} \big(c^w u_{jk} + \Delta_{jk}\varphi_{jk}\big)\right] \\
\textit{s.t.} \quad & \text{(8a)–(8f)},\ \text{(9d)},\ \text{(11)},\ \text{(12a)–(12b)}, \\
& \theta \geq \pi_{\mathtt{S}} - \pi_{\mathtt{T}}, \\
& \pi_{\mathtt{S}} - \pi_{\widehat{t}_1} \geq \widehat{t}_1 \eta_1^x + (1-\widehat{t}_1)(\zeta_1 + \eta_1^e), \quad \forall \widehat{t}_1 \in \mathbb{B}, \\
& \pi_{(\widehat{t}_1, \widehat{t}_{i-1})} - \pi_{(\widehat{t}_1, \widehat{t}_i)} \geq \eta_i^x + (1-\widehat{t}_i)(\zeta_i + \eta_i^e - \eta_i^x) \\
& \qquad\qquad + (1-\widehat{t}_{i-1})(1-\widehat{t}_i)\phi_{i-1}, \quad \forall \widehat{t}_{i-1}, \widehat{t}_i \in \mathbb{B},\ \forall i \in [2, I]_{\mathbb{Z}}, \\
& \pi_{(\widehat{t}_1, \widehat{t}_I)} - \pi_{\mathtt{T}} \geq (1-\widehat{t}_I)(1-\widehat{t}_1)\phi_I, \quad \forall \widehat{t}_1, \widehat{t}_I \in \mathbb{B}.
\end{aligned}$$

5 Optimal Nurse Pool Design

Of all the three practical nurse pool structures, Structure [1] is most flexible as every pool nurse is capable of working in all units. However, this incurs a high need for cross-training. For example, to enable a nurse working in a unit to be a pool nurse, he/she needs to be cross-trained for all the remaining $J-1$ units. As a result, enabling all nurses needs as many as $J(J-1)/2$ pairs of cross-training. In contrast, Structure [C] needs J pairs of cross-training because every pool consists of exactly two units. Structure [D] needs even less cross-training if we adopt a "sparse" design, e.g., pooling together a small subset of units. In this section, we examine how to design a sparse but effective pool structure that is disjoint. Specifically, we search for a disjoint pool structure that needs as few cross-training as possible, while achieving a pre-specified performance guarantee in terms of DR staffing cost.[2] To this end, we define binary variables a_{ij} such that $a_{ij} = 1$ if unit j is assigned to pool i and $a_{ij} = 0$ otherwise, binary variables o_i such that $o_i = 1$ if any units are assigned to pool i (i.e., if pool i is "opened") and $o_i = 0$ otherwise, and binary variables p_{jk} such

[2] We notice that there exist multiple alternative quantities that can be used to quantify the effort of cross-training. In this paper, we pick the number of pairs of cross-training as a representative objective function. Alternative objectives can be similarly modeled and computed.

that $p_{jk} = 1$ if units j and k are assigned to the same pool and $p_{jk} = 0$ otherwise. Then, the total amount of needed cross-training equals $\sum_{j=1}^{J}\sum_{k=j+1}^{J} p_{jk}$. In addition, these binary variables satisfy the following constraints:

$$\sum_{i=1}^{I+1} a_{ij} = 1, \quad \forall j \in [J], \tag{16a}$$

$$a_{ij} \le o_i, \quad \forall i \in [I],\ \forall j \in [J], \tag{16b}$$

$$p_{jk} \ge a_{ij} + a_{ik} - 1, \quad \forall i \in [I],\ \forall j,k \in [J] \text{ and } j < k, \tag{16c}$$

where constraints (16a) designate that each unit is assigned to one and only one pool (we create a dummy pool $I+1$ that collects all units that are not covered by any existing pools), constraints (16b) ensure that no units can be assigned to a pool if it is not opened, and constraints (16c) designate that $p_{jk} = 1$ if there is a pool i such that $a_{ij} = a_{ik} = 1$. If no such a pool i exists, then constraints (16b) reduce to $p_{jk} \ge 0$ and p_{jk} equals zero at optimality due to the objective function (17a). Based on Proposition 4, the optimal nurse pool design (OPD) model is formulated as

$$(\textbf{OPD}): \quad \min \ \sum_{j=1}^{J}\sum_{k=j+1}^{J} p_{jk} \tag{17a}$$

$$\text{s.t.} \quad \text{(8a)–(8f), (9d), (11), (12a)–(12b), (16a)–(16c),} \tag{17b}$$

$$\sum_{i=1}^{I}\Big(\theta_i + c^{\mathrm{y}} y_i^{\mathrm{L}} + g_i(y_i^{\mathrm{L}})\lambda_i + \sum_{\ell=1}^{y_i^{\mathrm{U}}-y_i^{\mathrm{L}}} \big(\delta_{i\ell}\nu_{i\ell} + c^{\mathrm{y}} v_{i\ell}\big)\Big) o_i + \sum_{j=1}^{J}\left[\sum_{q=1}^{Q} \mu_{jq}\rho_{jq} + c^{\mathrm{w}} w_j^{\mathrm{L}} + f_j(w_j^{\mathrm{L}})\gamma_j + \sum_{k=1}^{w_j^{\mathrm{U}}-w_j^{\mathrm{L}}} \big(c^{\mathrm{w}} u_{jk} + \Delta_{jk}\varphi_{jk}\big)\right] \le T, \tag{17c}$$

$$\theta_i \ge \phi_i + \sum_{\ell=1}^{J} (\zeta_\ell + \eta_\ell^{\mathrm{e}}) a_{i\ell}, \quad \forall i \in [I], \tag{17d}$$

$$\theta_i \ge \eta_j^{\mathrm{x}} a_{ij} + \sum_{\ell=1}^{j-1} \chi_\ell a_{i\ell} + \sum_{\ell=j+1}^{J} (\zeta_\ell + \eta_\ell^{\mathrm{e}}) a_{i\ell}, \quad \forall i \in [I],\ \forall j \in [J], \tag{17e}$$

where constraint (17c) ensures that the DR staffing cost does not exceed a given target T. If $y_i^{\mathrm{L}} = 0$ for all $i \in [I]$, i.e., if there is no minimum staffing requirement for pool nurses, then we shall pick T from the interval $[Z^\star_{[1]}, Z^\star_{[0]}]$, where $Z^\star_{[1]}$ represents the DR staffing cost with maximum flexibility and $Z^\star_{[0]}$ represents that with minimum flexibility. By gradually decreasing this target from $Z^\star_{[0]}$ to $Z^\star_{[1]}$, the amount of cross-training grows and accordingly we obtain a cost-training frontier that can clearly illustrate the trade-off between these two performance measures (see Section 6.4 for the numerical demonstration).

To effectively solve the (OPD) model, we recast it as a MILP in the following proposition.

Proposition 6 *Under Assumption 1, the (OPD) model* (17) *yields the same optimal objective value and the same set of optimal solutions as the following MILP:*

$$
\begin{aligned}
\min\ & \sum_{j=1}^{J}\sum_{k=j+1}^{J} p_{jk} \\
\text{s.t.}\ & \text{(8a)–(8f)},\ \text{(9d)},\ \text{(11)},\ \text{(12b)},\ \text{(16a)–(16c)}, \\
& \sum_{i=1}^{I}\Big(\theta_i + c^y y_i^L o_i + g_i(y_i^L)\lambda_i + \sum_{\ell=1}^{y_i^U-y_i^L}\big(\delta_{i\ell}\nu_{i\ell} + c^y v_{i\ell}\big)\Big) \\
& + \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + c^w w_j^L + f_j(w_j^L)\gamma_j + \sum_{k=1}^{w_j^U-w_j^L}\big(c^w u_{jk} + \Delta_{jk}\varphi_{jk}\big)\right] \ \le\ T, \\
& \left.\begin{array}{l}
\theta_i \ge \eta^x_{ij} + \sum_{\ell=1}^{j-1}\chi_{i\ell} + \sum_{\ell=j+1}^{J}(\zeta_{i\ell} + \eta^e_{i\ell}) \\
\chi_{ij} \ge \zeta_{ij} + \eta^e_{ij}, \quad \chi_{ij} \ge \eta^x_{ij} \\
0 \le \zeta_{ij} \le K a_{ij}, \quad -K a_{ij} \le \eta^x_{ij} \le K a_{ij}, \quad -K a_{ij} \le \eta^e_{ij} \le K a_{ij}
\end{array}\right\} \ \forall i \in [I+1],\ \forall j \in [J], \\
& \left.\begin{array}{l}
\theta_i \ge \phi_i + \sum_{\ell=1}^{J}(\zeta_{i\ell} + \eta^e_{i\ell}) \\
\phi_i \ge -c^e y_i^L o_i - y_i^L\lambda_i - \sum_{\ell=1}^{y_i^U-y_i^L}(\nu_{i\ell} + c^e v_{i\ell}) \\
-c^x o_i \le \lambda_i \le 0, \quad v_{i1} \le o_i
\end{array}\right\} \ \forall i \in [I+1], \\
& \zeta_j = \sum_{i=1}^{I+1}\zeta_{ij}, \quad \eta^x_j = \sum_{i=1}^{I+1}\eta^x_{ij}, \quad \eta^e_j = \sum_{i=1}^{I+1}\eta^e_{ij}, \quad \forall j \in [J],
\end{aligned}
$$

where K represents a sufficiently large positive constant.

In addition, the above formulation involves symmetric binary solutions. In Appendix O, we derive symmetry breaking inequalities to enhance its computational efficacy.

6 Numerical Experiments

In this section, we report numerical experiments on (DRNS) and (OPD) models. We summarize our main findings as follows:

1. Under the practical nurse pool structures as introduced in Section 4, the monolithic MILP reformulations of (DRNS) lead to significant speed-up over the separation algorithm.

2. Modeling nurse absenteeism improves the out-of-sample performance of staffing decisions. The improvement becomes more significant as the value of operational flexibility increases.

3. Even a very sparse nurse pool design can harvest most of the operational flexibility.

4. An optimal nurse pool design tends to pool together the units with higher variability, e.g., higher standard deviation of nurse demand and/or higher absence rate. In particular, the variability of nurse absenteeism plays a more important role in optimal pool design.

In all experiments, we solve optimization models by GUROBI 7.0.1 via Python 2.7 on a personal laptop with an Intel(R) Core(TM) i7-4850HQ CPU@2.3GHz and 16GB RAM.

6.1 Instance design

We design test instances based on the data and insights provided by our collaborating hospital and existing literature [28]. Specifically, we set $Q = 2$ for nurse demand uncertainty. That is, we consider the nurse demand mean value μ_{j1}, which is randomly extracted from the interval $[5, 20]$, and the standard deviation sd_j, which is randomly extracted from the interval $[0, 20]$. In addition, we assume a constant nurse absence rate such that $f_j(w_j) = A^{\text{u}}_j w_j$ and $g_i(y_i) = A^{\text{p}}_i y_i$, where A^{u}_j denotes absence rate of unit nurses and is randomly extracted from the interval $[0.60, 0.98]$ and A^{p}_i denotes that of pool nurses and is randomly extracted from the interval $[0.98, 1.00]$. For (DRNS), we set $w^{\text{L}}_j = \lfloor \underline{S} A^{\text{u}}_j \mu_{j1} \rfloor$, $w^{\text{U}}_j = 200$ for all $j \in [J]$ and $y^{\text{L}}_i = 0$, $y^{\text{U}}_i = 200$ for all $i \in [I]$, where $\underline{S}$ represents a safety constant. In practice, a positive w^{L}_j helps to maintain a constant roster in each unit to promote teamwork. We also incorporate an integrative staffing upper bound by specifying that

$$R := \left\{ (w, y) : \sum_{j=1}^{J} w_j + \sum_{i=1}^{I} y_i \leq \left\lceil \overline{S} \sum_{j=1}^{J} \left(\frac{\sum_{j=1}^{J} A^{\text{u}}_j + \sum_{i=1}^{I} A^{\text{p}}_i}{I + J} \right) \mu_{j1} \right\rceil \right\},$$

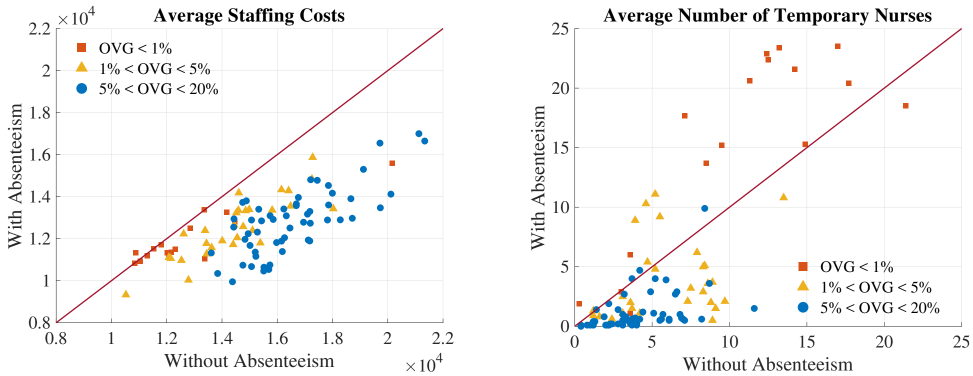
where $\overline{S}$ represents another safety constant that describes an upper limit on the human resource.

Table 1: Average wall-clock seconds used to solve (DRNS)

$[I, J]$	Separation	MILP	$[I, J]$	Separation	MILP
$[1, 5]$	2.28	0.09	$[3, 5]$	1.93	0.09
$[1, 7]$	8.24	0.10	$[3, 7]$	9.82	0.10
$[1, 10]$	44.42	0.13	$[3, 10]$	68.33	0.16
$[1, 20]$	> 3600	0.37	$[3, 20]$	> 3600	0.37
$[1, 50]$	> 3600	1.51	$[3, 50]$	> 3600	1.13

6.2 Computational performance

We compare the computational efficacy of the separation algorithm and the monolithic MILP reformulations on solving (DRNS) under practical pool structures. Specifically, we create 10 random test instances with various $[I, J]$ combinations, where $I = 1$ indicates one single pool (i.e., Structure [1]) and $I = 3$ indicates three disjoint pools (e.g., Structure [D]). We report the computing time (in wall-clock seconds) in Table 1. From this table, we observe that the time spent by the separation algorithm quickly increases and hits the 1-hour time limit as J increases. In contrast, the MILP reformulations are significantly more scalable and can be solved to global optimality within 2 seconds in all instances.



(a) Average staffing costs

(b) Average number of temporary nurses

Figure 4: An out-of-sample comparison of considering versus overlooking nurse absenteeism

6.3 Value of modeling nurse absenteeism

As discussed in Section 1, modeling nurse absenteeism incurs endogenous uncertainty and computational challenges. It is hence worth examining what (DRNS) buys us, i.e., the value of modeling nurse absenteeism. To this end, we consider a test instance with 7 units and one single pool (i.e., under Structure [1]). In addition, we consider a variant of (DRNS) that overlooks the nurse absenteeism, in which we assume that all assigned nurses show up. Then, we compare the out-of-sample performance of the optimal nurse staffing decisions produced by (DRNS) and that produced by overlooking absenteeism. Fixing the nurse staffing levels as in a (DRNS) optimal solution (w^*, y^*), we generate a large number of scenarios for nurse demand and absenteeism, where the demands follow log-normal distribution, i.e., $\tilde{d}_j \sim \text{LN}(\mu_{j1}, \text{sd}_j)$, and the numbers of present nurses follow binomial distribution, i.e., $\tilde{w}_j \sim B(w^*_j, A^{\text{u}}_j)$ and $\tilde{y}_i \sim B(y^*_i, A^{\text{p}}_i)$. Exposing (w^*, y^*) under these scenarios produces an out-of-sample estimate of the average staffing cost with absenteeism, which we denote by Z_{abs}. Using the same set of scenarios, we examine the optimal solution produced by overlooking absenteeism and obtain an out-of-sample average cost without absenteeism, denoted by $Z_{\text{w/o}}$. Using the same out-of-sample procedures, we compute the average number of temporary nurses hired when considering absenteeism (denoted by x_{abs}) and when overlooking it (denoted by $x_{\text{w/o}}$).

We depict the values of $Z_{\text{w/o}}$ (x-coordinate) and Z_{abs} (y-coordinate) obtained in 100 replications in Figure 4a. From this figure, we observe that most dots are below the 45-degree line, indicating that $Z_{\text{w/o}} - Z_{\text{abs}} > 0$, i.e., modeling nurse absenteeism yields nurse staffing levels with better out-of-sample performance. In addition, we group the dots based on the relative value of operational flexibility OVG $:= (Z^\star_{[0]} - Z^\star_{[1]})/Z^\star_{[0]} \times 100\%$. From Figure 4a, we observe that the difference $Z_{\text{w/o}} - Z_{\text{abs}}$ shows an increasing trend as OVG increases. That is, modeling nurse absenteeism becomes more valuable as the value of operational flexibility increases. This makes sense because when a unit is short of supply due to nurse absenteeism, making it up with pool nurses are less expensive than doing so with temporary nurses. As a result, setting up nurse pools can effectively mitigate the impacts of nurse absenteeism. In Figure 4b, we depict the values of $x_{\text{w/o}}$ and x_{abs} obtained in the 100 replications and make similar observations.

6.4 Comparison among various pool structures

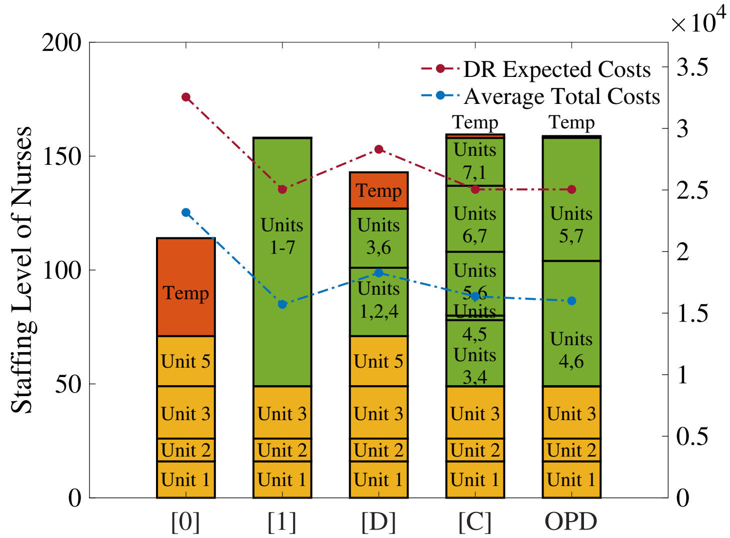


Figure 5: Staffing levels and out-of-sample performance under various pool structures

We compare the operational cost of nurse staffing under Structures [0], [1], [D], [C], and under the optimal pool design obtained from the (OPD) model. To this end, we generate a set of random test instances and solve each instance under all the structures. Then, by fixing the nurse staffing levels at the obtained optimal solution under each structure, we conduct an out-of-sample simulation to compute the average staffing cost of each solution based on scenarios of nurse demand and absenteeism. In this experiment, we observe that a sparse nurse pool design can often achieve similar out-of-sample performance as under Structure [1]. We report the input parameters of a representative instance in Appendix P and the results of this instance in Figure 5. From this figure, we observe that the total number of unit nurses and that of pool nurses hired under Structures [1], [C], and (OPD) are similar. Likewise, their DR staffing costs and out-of-sample average staffing cost are close. Nevertheless, these structures are drastically different in the amount of cross-training. For example, Structures [1] and (OPD) cross-train 21 and 2 pairs of units, respectively. That is, by cross-training two pairs of units, the optimal pool design produced by (OPD) harvests nearly all the operational flexibility of cross-training all the units.

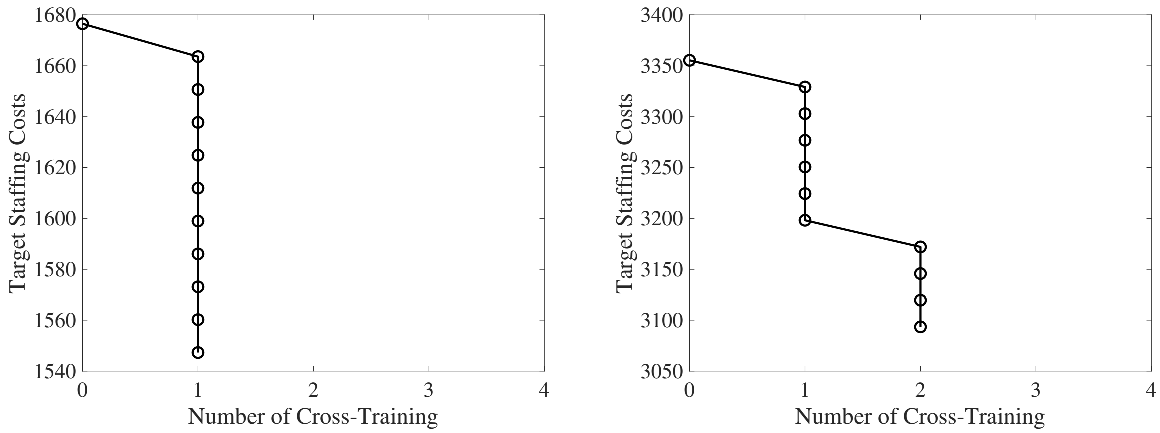


Figure 6: Amount of cross-training as a function of the target operational cost

To further verify this observation, we generate another set of random test instances and conduct sensitivity analysis on the target staffing cost T in the (OPD) model (the input parameters are specified later in Section 6.5). Specifically, we uniformly pick ten values of T between $Z^{\star}_{[0]}$ (i.e., the optimal value of (DRNS) with no nurse pools) and $Z^{\star}_{[1]}$ (i.e., the optimal value of (DRNS) under Structure [1]). For each value of T, we solve (OPD) to obtain the minimum amount of cross-training $\#(T)$ that guarantees that the DR staffing cost is no larger than T. We report the curve of $\#(T)$ in two representative instances in Figure 6. The results confirm our observations from Figure 5.

6.5 Patterns of the Optimal Nurse Pool Design

We notice from Figure 5 that a sparse pool design does not simply yield good out-of-sample performance. For example, in this figure, Structure [D] pools together units 3, 6 and units 1, 2, 4, and yields a considerably higher out-of-sample average staffing cost than that of (OPD), which pools together units 4, 6 and units 5, 7. From the input parameters of this instance (see Appendix P), we observe that units 1, 2, and 3 have lower variability in nurse demand and lower nurse absence rate, while the remaining units are more variable in both nurse demand and absenteeism. We hence conjecture that an optimal design tends to pool together units with higher variability (i.e., higher standard deviation in demand and/or higher absence rate). We numerically verify this conjecture in the following experiments.

We generate a set of random test instances with 4 or 8 units, $c^{\mathrm{x}}/c^{\mathrm{w}} = 2$, $c^{\mathrm{y}}/c^{\mathrm{w}} = 1.1$, $A^{\mathrm{p}}_i = 0.99$, $\overline{S} \in \{0.5, 1.0, 1.5\}$, and $\underline{S} \in \{0.1, 0.2, 0.3\}$. In addition, we divide the units into two disjoint subsets $\mathcal{A}$ and $\mathcal{B}$, where units in $\mathcal{A}$ have lower variability and those in $\mathcal{B}$ have higher variability. We consider the following three cases depending on what variability refers to:

Table 2: Three cases on constructing subsets $\mathcal{A}$ and $\mathcal{B}$

	Units in $\mathcal{A}$	Units in $\mathcal{B}$
Case 1	Low sd_j, Low A^{u}_j	High sd_j, Low A^{u}_j
Case 2	Low sd_j, Low A^{u}_j	Low sd_j, High A^{u}_j
Case 3	Low sd_j, Low A^{u}_j	High sd_j, High A^{u}_j

Note that (i) a value of low (respectively, high) standard deviation of nurse demand is randomly extracted from the interval $[7.24, 7.92]$ (respectively, $[17.14, 18.42]$), (ii) a value of low (respectively, high) absence rate is randomly extracted from the interval $[0.02, 0.04]$ (respectively, $[0.20, 0.40]$), and (iii) the mean value of nurse demand is randomly extracted from the interval $[25, 27]$. Finally, we set $T = Z^{\star}_{[1]}$, i.e., we are interested in the most sparse pool structures that produce equally good DR staffing cost as under Structure [1].

Table 3: OPD and OVG (%) for the instances with $\overline{S} = 1.5$ and $\underline{S} = 0.1$

4-Unit System			8-Unit System		
Case 1	#Pools for each type	OVG (%)	Case 1	#Pools for each type	OVG (%)
Instance 1-1	[0,0,2]	14.17	Instance 1-1	[1,1,2]	13.45
Instance 1-2	[1,1,0]	14.36	Instance 1-2	[1,1,2]	14.72
Instance 1-3	[0,0,2]	15.08	Instance 1-3	[1,1,2]	13.42
Instance 1-4	[1,1,0]	14.58	Instance 1-4	[1,1,2]	14.35
Instance 1-5	[0,0,2]	13.92	Instance 1-5	[1,1,2]	14.07
Case 2	#Pools for each type	OVG (%)	Case 2	#Pools for each type	OVG (%)
Instance 2-1	[0,1,0]	20.04	Instance 2-1	[0,2,0]	15.48
Instance 2-2	[0,1,0]	19.68	Instance 2-2	[0,2,0]	16.67
Instance 2-3	[0,1,0]	15.14	Instance 2-3	[0,2,0]	17.48
Instance 2-4	[0,1,0]	15.86	Instance 2-4	[0,2,0]	17.11
Instance 2-5	[0,1,0]	20.86	Instance 2-5	[0,2,0]	16.06
Case 3	#Pools for each type	OVG (%)	Case 3	#Pools for each type	OVG (%)
Instance 3-1	[0,1,0]	11.98	Instance 3-1	[0,2,0]	11.55
Instance 3-2	[0,1,0]	12.89	Instance 3-2	[0,2,0]	12.21
Instance 3-3	[0,1,0]	11.01	Instance 3-3	[0,2,0]	11.43
Instance 3-4	[0,1,0]	11.23	Instance 3-4	[0,2,0]	12.69
Instance 3-5	[0,1,0]	11.37	Instance 3-5	[0,2,0]	12.07

We classify the pools produced by (OPD) into three types based on the variability of the units a pool covers. We call a pool "Type-1" if all the units in this pool come from subset $\mathcal{A}$, "Type-2" if all the units come from $\mathcal{B}$, and "Type-3" if the units come from both $\mathcal{A}$ and $\mathcal{B}$. We report the frequencies of each type appearing in an optimal pool design and the corresponding OVG in Table 3.[3] Take Instance 1-1 in Case 1 with the 8-unit system for example. The optimal design of this instance (see $[1, 1, 2]$ in the fifth column) consists of one Type-1 pool, one Type-2 pool, and two Type-3 pools. From this table, we observe that the optimal pool design diversifies among all three types in Case 1, i.e., the pools include both units with low variability and those with high variability. In contrast, in Cases 2 and 3, Type-2 pools become dominant, i.e., a majority of the pools include only the units with high variability. This observation numerically confirms our conjecture.

[3] The results in this table are associated with $\overline{S} = 1.5$ and $\underline{S} = 0.1$, but the observations remain the same for all other $\overline{S}$, $\underline{S}$ combinations.

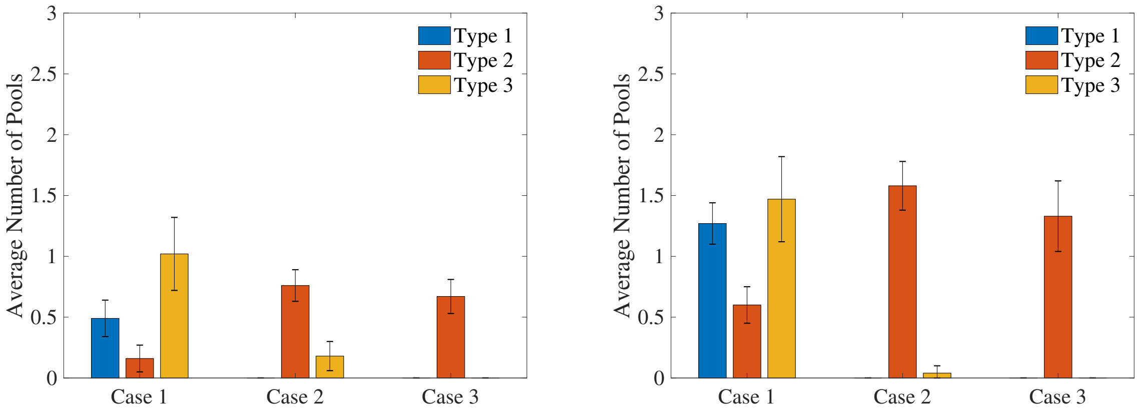


Figure 7: Types of the optimal pool design

We report the average number of each type appearing in an optimal pool design among all instances in Figure 7, where the error bars represent the corresponding 80%-confidence interval. From this figure, we observe that the Type-2 pools become dominant as we move to Cases 2 or 3. This once again confirms our conjecture numerically. In addition, we notice that the dominance of the Type-2 pools vanishes when moving from Case 3 to Case 1, i.e., when the variability of absenteeism decreases and that of demand remains unchanged. In contrast, the dominance of the Type-2 pools stays the same when moving from Case 3 to Case 2, i.e., when the variability of absenteeism remains unchanged and that of demand decreases. This indicates that the variability of nurse absenteeism plays a more important role in deciding the pattern of the optimal pool design. Hence, this result suggests that we should prioritize pooling together the units with higher variability, and especially those with higher nurse absence rates.

7 Conclusions

We studied a two-stage (DRNS) model for nurse staffing under both exogenous demand uncertainty and endogenous absenteeism uncertainty. We derived a min-max reformulation for (DRNS) under arbitrary nurse pool structures, leading to a separation algorithm that provably finds a globally optimal solution within a finite number of iterations. Under practical pool structures including one pool, disjoint pools, and chained pools, we derived monolithic MILP reformulations for (DRNS) and significantly improved the computational efficacy. Via numerical case studies, we found that modeling absenteeism improves the out-of-sample performance of staffing decisions, and such improvement is positively correlated with the value of operational flexibility. For nurse pool design, we found that sparse pool structures can already harvest most of the operational flexibility. More importantly, it is particularly effective to pool together the units with higher nurse absence rates.

Appendix A Proof of Lemma 1

Proof: Representing variables $e_j \equiv \sum_{i:j\in P_i} z_{ij} + x_j - \tilde{d}_j + \tilde{w}_j$ by constraints (1b), we rewrite formulation (1a)–(1d) as

$$
\begin{aligned}
V(\tilde{w}, \tilde{y}, \tilde{d}) \;=\; \min_{z,x} \;\; & \sum_{j=1}^{J}\Big[(c^{\mathrm{x}} - c^{\mathrm{e}})x_j - c^{\mathrm{e}} \sum_{i:j\in P_i} z_{ij}\Big] + \sum_{j=1}^{J} c^{\mathrm{e}}(\tilde{d}_j - \tilde{w}_j) \\
\text{s.t.} \;\; & x_j + \sum_{i:j\in P_i} z_{ij} \geq \tilde{d}_j - \tilde{w}_j, \quad \forall j \in [J], && (18a) \\
& \sum_{j\in P_i} z_{ij} \leq \tilde{y}_i, \quad \forall i \in [I], && (18b) \\
& x_j \in \mathbb{Z}_+, \quad \forall j \in [J], \quad z_{ij} \in \mathbb{Z}_+, \quad \forall i \in [I], \; \forall j \in P_i.
\end{aligned}
$$

We note that the constraint matrix of the above formulation is totally unimodular (TU), and so the conclusion follows. To see the TU property, we multiply each of the constraints (18b) by -1 on both sides and recast the constraint matrix of (18a)–(18b) in the following form:

$$
\begin{bmatrix} \Big(x_j + \sum_{i:j\in P_i} z_{ij}\Big) \\ \Big(-\sum_{j\in P_i} z_{ij}\Big) \end{bmatrix}.
$$

It follows that (a) each entry of this matrix is -1, 0, or 1, (b) this matrix has at most two nonzero entries in each column, and (c) the entries sum up to be zero for any column containing two nonzero entries. Hence, the constraint matrix is TU based on Proposition 2.6 in [31]. The conclusion follows because $\tilde{d}_j - \tilde{w}_j$ and $-\tilde{y}_i$ are integers for all $j \in [J]$ and for all $i \in [I]$, respectively. □

Appendix B Verifying Assumption 1

We present necessary and sufficient conditions for Assumption 1 in the following proposition.

Proposition 7 *For any given w and y, $\mathcal{D}$ is non-empty if and only if the following three conditions are satisfied:*

1. $f_j(w_j) \in [0, w_j]$ *for all* $j \in [J]$*;*

2. $g_i(y_i) \in [0, y_i]$ *for all* $i \in [I]$*;*

3. *For all $j \in [J]$, the optimal value of the following linear program is non-positive:*

$$
\min_{p_j \geq 0, \tau \geq 0} \;\; \sum_{q=1}^{Q} (\tau_q^+ + \tau_q^-) \tag{19a}
$$

$$s.t. \quad \sum_{k=d_j^L}^{d_j^U} k^q p_{jk} + \tau_q^+ - \tau_q^- = \mu_{jq}, \quad \forall q \in [Q], \tag{19b}$$

$$\sum_{k=d_j^L}^{d_j^U} p_{jk} = 1. \tag{19c}$$

Proof: (**Necessity**) Suppose that $\mathcal{D} \neq \emptyset$. Then, there exists a $\mathbb{P} \in \mathcal{P}(\Xi)$ such that $\mathbb{E}_{\mathbb{P}}[\tilde{d}_j^q] = \mu_{jq}$ for all $j \in [J]$ and $q \in [Q]$, $\mathbb{E}_{\mathbb{P}}[\tilde{w}_j] = f_j(w_j)$ for all $j \in [J]$, and $\mathbb{E}_{\mathbb{P}}[\tilde{y}_i] = g_i(y_i)$ for all $i \in [I]$. It follows that, for all $j \in [J]$, we have $f_j(w_j) \leq \text{esssup}_{\Xi}\{\tilde{w}_j\} \leq w_j$ and $f_j(w_j) \geq \text{essinf}_{\Xi}\{\tilde{w}_j\} \geq 0$, leading to $f_j(w_j) \in [0, w_j]$. Likewise, it holds that $g_i(y_i) \in [0, y_i]$ for all $i \in [I]$. In addition, for all $j \in [J]$ and $k \in [d_j^{\text{L}}, d_j^{\text{U}}]_{\mathbb{Z}}$, we let $\bar{p}_{jk} = \mathbb{P}\{\tilde{d}_j = k\}$. It follows that, for all $q \in [Q]$, $\sum_{k=d_j^{\text{L}}}^{d_j^{\text{U}}} \bar{p}_{jk} = \sum_{k=d_j^{\text{L}}}^{d_j^{\text{U}}} \mathbb{P}\{\tilde{d}_j = k\} = 1$. Moreover,

$$\sum_{k=d_j^{\text{L}}}^{d_j^{\text{U}}} k^q \bar{p}_{jk} \;=\; \sum_{k=d_j^{\text{L}}}^{d_j^{\text{U}}} k^q \mathbb{P}\{\tilde{d}_j = k\} \;=\; \mathbb{E}_{\mathbb{P}}[\tilde{d}_j^q] \;=\; \mu_{jq}.$$

Hence, together with $\tau_q^+ = \tau_q^- = 0$, $\bar{p}_{jk}$ constitutes a feasible solution to linear program (19) with an objective value being zero. As zero is also a lower bound of the objective value, $\bar{p}_{jk}$ is optimal to (19) and accordingly the optimal value of this linear program equals zero. This holds for all $j \in [J]$ and proves the necessity of the three conditions.

(**Sufficiency**) Suppose that the three conditions are satisfied. For all $j \in [J]$, as $f_j(w_j) \in [0, w_j] \equiv \text{conv}([0, w_j]_{\mathbb{Z}})$ by condition 1, there exists a $\mathbb{P}_{\tilde{w}_j} \in \mathcal{P}([0, w_j]_{\mathbb{Z}})$ such that $f_j(w_j) = \mathbb{E}_{\mathbb{P}_{\tilde{w}_j}}[\tilde{w}_j]$. Likewise, for all $i \in [I]$, there exists a $\mathbb{P}_{\tilde{y}_i} \in \mathcal{P}([0, y_i]_{\mathbb{Z}})$ such that $g_i(y_i) = \mathbb{E}_{\mathbb{P}_{\tilde{y}_i}}[\tilde{y}_i]$. In addition, as the optimal value of (19) is non-positive and $\tau_q^+, \tau_q^- \geq 0$ for all $q \in [Q]$, the optimal value of (19) equals zero. It follows that, for all $j \in [J]$, there exist p_{jk} such that $\sum_{k=d_j^{\text{L}}}^{d_j^{\text{U}}} k^q p_{jk} = \mu_{jq}$ for all $q \in [Q]$ and $\sum_{k=d_j^{\text{L}}}^{d_j^{\text{U}}} p_{jk} = 1$. Defining $\mathbb{P}_{\tilde{d}_j} \in \mathcal{P}([d_j^{\text{L}}, d_j^{\text{U}}]_{\mathbb{Z}})$ such that $\mathbb{P}_{\tilde{d}_j}\{\tilde{d}_j = k\} = p_{jk}$ for all $k \in [d_j^{\text{L}}, d_j^{\text{U}}]_{\mathbb{Z}}$, we have $\mathbb{E}_{\mathbb{P}_{\tilde{d}_j}}[\tilde{d}_j^q] = \mu_{jq}$. Therefore, the probability distribution

$$\mathbb{P} \;:=\; \Pi_{j=1}^J \mathbb{P}_{\tilde{w}_j} \times \Pi_{i=1}^I \mathbb{P}_{\tilde{y}_i} \times \Pi_{j=1}^J \mathbb{P}_{\tilde{d}_j}$$

satisfies constraints (2a)–(2c) and hence $\mathbb{P} \in \mathcal{D}$. It follows that $\mathcal{D} \neq \emptyset$ and the proof is completed. □

Appendix C Proof of Proposition 1

Proof: First, denoting $\tilde{\xi} := (\tilde{w}, \tilde{y}, \tilde{d})$, we present $\sup_{\mathbb{P} \in \mathcal{D}} \mathbb{E}_{\mathbb{P}}[V(\tilde{w}, \tilde{y}, \tilde{d})]$ as the following optimization problem:

$$\max_{p \geq 0} \quad \sum_{\tilde{\xi} \in \Xi} p_{\tilde{\xi}} V(\tilde{\xi})$$

$$\text{s.t.} \quad \sum_{\tilde{\xi}\in\Xi} p_{\tilde{\xi}}\tilde{w}_j = f_j(w_j), \quad \forall j \in [J], \tag{20a}$$

$$\sum_{\tilde{\xi}\in\Xi} p_{\tilde{\xi}}\tilde{y}_i = g_i(y_i), \quad \forall i \in [I], \tag{20b}$$

$$\sum_{\tilde{\xi}\in\Xi} p_{\tilde{\xi}}\tilde{d}_j^q = \mu_{jq}, \quad \forall j \in [J],\ \forall q \in [Q], \tag{20c}$$

$$\sum_{\tilde{\xi}\in\Xi} p_{\tilde{\xi}} = 1, \tag{20d}$$

where decision variables $p_{\tilde{\xi}}$ represent the probability of the random variables being realized as $\tilde{\xi}$, and constraints (20a)–(20d) describe the ambiguity set $\mathcal{D}$ defined in (2a)–(2c). The dual of this formulation is

$$\min_{\gamma,\lambda,\rho,\theta} \quad \sum_{j=1}^{J}\sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + \sum_{j=1}^{J} f_j(w_j)\gamma_j + \sum_{i=1}^{I} g_i(y_i)\lambda_i + \theta \tag{21a}$$

$$\text{s.t.} \quad \theta + \sum_{j=1}^{J}\sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q + \sum_{j=1}^{J}\gamma_j\tilde{w}_j + \sum_{i=1}^{I}\lambda_i\tilde{y}_i \geq V(\tilde{\xi}), \quad \forall \tilde{\xi} \in \Xi, \tag{21b}$$

where dual variables γ_j, λ_i, ρ_{jq}, and θ are associated with primal constraints (20a)–(20d), respectively, and dual constraints (21b) are associated with primal variables $p_{\tilde{\xi}}$. By Assumption 1, strong duality holds between the primal and dual formulations because they are both linear programs. As the objective function aims to minimize the value of θ, we observe by constraints (21b) that $\theta = \sup_{\tilde{\xi}\in\Omega}\{V(\tilde{\xi}) - \sum_{j=1}^{J}\sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q - \sum_{j=1}^{J}\gamma_j\tilde{w}_j - \sum_{i=1}^{I}\lambda_i\tilde{y}_i\}$. Hence, $\sup_{\mathbb{P}\in\mathcal{D}}\mathbb{E}_{\mathbb{P}}[V(\tilde{\xi})]$ equals the optimal value of the following min-max optimization problem:

$$\min_{\gamma,\lambda,\rho}\max_{\tilde{\xi}\in\Xi}\left\{V(\tilde{\xi}) - \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q + \gamma_j\tilde{w}_j\right] - \sum_{i=1}^{I}\lambda_i\tilde{y}_i\right\}$$
$$+ \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\mu_{jq}\rho_{jq} + f_j(w_j)\gamma_j\right] + \sum_{i=1}^{I} g_i(y_i)\lambda_i. \tag{22a}$$

Second, in view of the dual formulation (4a)–(4c) of $V(\tilde{\xi})$, we rewrite the maximum term in (22a) as

$$\max_{(\tilde{w},\tilde{y},\tilde{d})\in\Xi}\ \max_{(\alpha,\beta)\in\Lambda}\left\{\sum_{j=1}^{J}(\tilde{d}_j - \tilde{w}_j)\alpha_j + \sum_{i=1}^{I}\tilde{y}_i\beta_i - \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q + \gamma_j\tilde{w}_j\right] - \sum_{i=1}^{I}\lambda_i\tilde{y}_i\right\}$$
$$= \max_{(\alpha,\beta)\in\Lambda}\ \max_{(\tilde{w},\tilde{y},\tilde{d})\in\Xi}\left\{\sum_{j=1}^{J}(\tilde{d}_j - \tilde{w}_j)\alpha_j + \sum_{i=1}^{I}\tilde{y}_i\beta_i - \sum_{j=1}^{J}\left[\sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q + \gamma_j\tilde{w}_j\right] - \sum_{i=1}^{I}\lambda_i\tilde{y}_i\right\}$$
$$= \max_{(\alpha,\beta)\in\Lambda}\left\{\sum_{j=1}^{J}\max_{\tilde{w}_j\in[0,w_j]_{\mathbb{Z}}}\Big\{(-\alpha_j - \gamma_j)\tilde{w}_j\Big\} + \sum_{i=1}^{I}\max_{\tilde{y}_i\in[0,y_i]_{\mathbb{Z}}}\Big\{(\beta_i - \lambda_i)\tilde{y}_i\Big\} + \sum_{j=1}^{J}\max_{\tilde{d}_j\in[d_j^{\mathrm{L}},d_j^{\mathrm{U}}]_{\mathbb{Z}}}\Big\{\alpha_j\tilde{d}_j - \sum_{q=1}^{Q}\rho_{jq}\tilde{d}_j^q\Big\}\right\}.$$

Finally, as $(-\alpha_j - \gamma_j)\tilde{w}_j$ is linear in $\tilde{w}_j$, we have

$$\max_{\tilde{w}_j \in [0, w_j]_{\mathbb{Z}}} \Big\{(-\alpha_j - \gamma_j)\tilde{w}_j\Big\} = \max\Big\{0, (-\alpha_j - \gamma_j)w_j\Big\} = \Big[(-\alpha_j - \gamma_j)w_j\Big]_+.$$

Similarly, we have $\max_{\tilde{y}_i \in [0, y_i]_{\mathbb{Z}}}\{(\beta_i - \lambda_i)\tilde{y}_i\} = [(\beta_i - \lambda_i)y_i]_+$. This completes the proof. □

Appendix D Proof of Lemma 2

Proof: As $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$ is to maximize a convex function over a polyhedron, we only need to analyze the extreme directions and extreme points of Λ.

First, the extreme directions of Λ are $(\alpha, \beta) = (0, -e_i)$ for all $i \in [I]$, where e_i represents the i^{th} standard basis vector. As $y_i \geq 0$, moving along any of these extreme directions (i.e., decreasing the value of any β_i) does not increase the value of $F(\alpha, \beta)$. Hence, we can omit these extreme directions in the attempt of maximizing $F(\alpha, \beta)$ and accordingly $\bar{\beta}_i = \min\{-\bar{\alpha}_j : j \in P_i\} = -\max\{\bar{\alpha}_j : j \in P_i\}$ without loss of optimality. This proves property (b) in the claim. In addition, there exists an extreme point of Λ that is optimal to $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$.

Second, we prove, by contradiction, that each extreme point of Λ satisfies property (a) in the claim. Suppose that there exists an extreme point $(\bar{\alpha}, \bar{\beta})$ such that property (a) fails, i.e., $\bar{\alpha}_{j^*} \in (c^{\text{e}}, c^{\text{x}})$ for some $j^* \in [J]$. Consider the set $\mathcal{I}(j^*) := \{i \in [I] : -\bar{\beta}_i = \bar{\alpha}_{j^*}\}$. We discuss the following two cases. In each case, we shall construct two points in Λ such that their midpoint is $(\bar{\alpha}, \bar{\beta})$, which provides a desired contradiction.

1. If $\mathcal{I}(j^*) = \emptyset$, then $-\bar{\beta}_i > \bar{\alpha}_{j^*}$ for all i such that $j^* \in P_i$. Defining $\epsilon := (1/2)\min\{-\bar{\beta}_i - \bar{\alpha}_{j^*}, \bar{\alpha}_{j^*} - c^{\text{e}}, c^{\text{x}} - \bar{\alpha}_{j^*}\} > 0$, we construct two points $(\bar{\alpha}^+, \bar{\beta})$ and $(\bar{\alpha}^-, \bar{\beta})$ such that $\bar{\alpha}^+_{j^*} = \bar{\alpha}_{j^*} + \epsilon$, $\bar{\alpha}^-_{j^*} = \bar{\alpha}_{j^*} - \epsilon$, and $\bar{\alpha}^+_j = \bar{\alpha}^-_j = \bar{\alpha}_j$ for all $j \neq j^*$. Then, it is clear that $(\bar{\alpha}^+, \bar{\beta}), (\bar{\alpha}^-, \bar{\beta}) \in \Lambda$. But $(\bar{\alpha}, \bar{\beta}) = (1/2)(\bar{\alpha}^+, \bar{\beta}) + (1/2)(\bar{\alpha}^-, \bar{\beta})$, which contradicts the fact that $(\bar{\alpha}, \bar{\beta})$ is an extreme point of Λ.

2. If $\mathcal{I}(j^*) \neq \emptyset$, then we define $\mathcal{J}(j^*) := \bigcup_{i\in\mathcal{I}(j^*)}\{j \in P_i : \bar{\alpha}_j = -\bar{\beta}_i\}$. It follows that $\bar{\alpha}_j = \bar{\alpha}_{j^*}$ for all $j \in \mathcal{J}(j^*)$. Hence, for each $i \in \mathcal{I}(j^*)$, $\bar{\alpha}_j = \bar{\alpha}_{j^*}$ for all $j \in P_i \cap \mathcal{J}(j^*)$ and $\bar{\alpha}_j < \bar{\alpha}_{j^*}$ for all $j \in P_i \backslash \mathcal{J}(j^*)$. We define $\epsilon := (1/2)\min\Big\{\min\{\bar{\alpha}_{j^*} - \bar{\alpha}_j : i \in \mathcal{I}(j^*), j \in P_i \backslash \mathcal{J}(j^*)\},\ \min\{-\bar{\beta}_i - \bar{\alpha}_{j^*} : i \notin \mathcal{I}(j^*),\ -\bar{\beta}_i > \bar{\alpha}_{j^*}\},\ \bar{\alpha}_{j^*} - c^{\text{e}}, c^{\text{x}} - \bar{\alpha}_{j^*}\Big\}$. Then $\epsilon > 0$ because it is the minimum of a finite number of positive reals.[4] We construct two points $(\bar{\alpha}^+, \bar{\beta}^+)$ and $(\bar{\alpha}^-, \bar{\beta}^-)$ such that

$$\bar{\alpha}^+_j = \begin{cases} \bar{\alpha}_{j^*} + \epsilon & \forall j \in \mathcal{J}(j^*) \\ \bar{\alpha}_j & \text{otherwise} \end{cases}, \quad \bar{\alpha}^-_j = \begin{cases} \bar{\alpha}_{j^*} - \epsilon & \forall j \in \mathcal{J}(j^*) \\ \bar{\alpha}_j & \text{otherwise} \end{cases},$$

[4]Here we adopt the convention that $\min\{a : a \in A\} = \infty$ if $A = \emptyset$. For example, if there does not exist an $i \notin \mathcal{I}(j^*)$ such that $-\bar{\beta}_i > \bar{\alpha}_{j^*}$, then $\min\{-\bar{\beta}_i - \bar{\alpha}_{j^*} : i \notin \mathcal{I}(j^*),\ -\bar{\beta}_i > \bar{\alpha}_{j^*}\} = \infty$.

$$\bar{\beta}_i^+ = \begin{cases} -(\bar{\alpha}_{j^*} + \epsilon) & \forall i \in \mathcal{I}(j^*) \\ \bar{\beta}_i & \text{otherwise} \end{cases}, \quad \bar{\beta}_i^- = \begin{cases} -(\bar{\alpha}_{j^*} - \epsilon) & \forall i \in \mathcal{I}(j^*) \\ \bar{\beta}_i & \text{otherwise} \end{cases}.$$

It is clear that $(\bar{\alpha}, \bar{\beta}) = (1/2)(\bar{\alpha}^+, \bar{\beta}^+) + (1/2)(\bar{\alpha}^-, \bar{\beta}^-)$. To finish the proof, it remains to show that $(\bar{\alpha}^+, \bar{\beta}^+), (\bar{\alpha}^-, \bar{\beta}^-) \in \Lambda$. To see this, we check constraints (4b) and (4c). For constraints (4c), we have $\bar{\alpha}_j^+ \in (c^{\mathrm{e}}, c^{\mathrm{x}})$ for all $j \in \mathcal{J}(j^*)$ by the definition of ϵ. Additionally, for all $j \notin \mathcal{J}(j^*)$, we have $\bar{\alpha}_j^+ = \bar{\alpha}_j^- = \bar{\alpha}_j \in [c^{\mathrm{e}}, c^{\mathrm{x}}]$. Hence, constraints (4c) are indeed satisfied and it remains to check constraints (4b). For each $i \in \mathcal{I}(j^*)$, $-\bar{\beta}_i^+ = \bar{\alpha}_{j^*} + \epsilon = \bar{\alpha}_j^+$ for all $j \in P_i \cap \mathcal{J}(j^*)$ and $-\bar{\beta}_i^+ = \bar{\alpha}_{j^*} + \epsilon \geq \bar{\alpha}_{j^*} \geq \bar{\alpha}_j = \bar{\alpha}_j^+$ for all $j \in P_i \setminus \mathcal{J}(j^*)$, where the first inequality is because $\epsilon > 0$, and the second inequality follows from the definition of $\mathcal{J}(j^*)$. Meanwhile, $-\bar{\beta}_i^- = \bar{\alpha}_{j^*} - \epsilon = \bar{\alpha}_j^-$ for all $j \in P_i \cap \mathcal{J}(j^*)$, and $-\bar{\beta}_i^- = \bar{\alpha}_{j^*} - \epsilon \geq \bar{\alpha}_j = \bar{\alpha}_j^-$ for all $j \in P_i \setminus \mathcal{J}(j^*)$, where the inequality follows from the definition of ϵ and the last equality is because $j \notin \mathcal{J}(j^*)$. It follows that constraints (4b) are indeed satisfied for all $i \in \mathcal{I}(j^*)$. For each $i \notin \mathcal{I}(j^*)$, $\bar{\beta}_i^+ = \bar{\beta}_i^- = \bar{\beta}_i$ and $-\bar{\beta}_i \neq \bar{\alpha}_{j^*}$. We discuss the following two sub-cases to complete the proof.

(a) If $-\bar{\beta}_i > \bar{\alpha}_{j^*}$, then $-\bar{\beta}_i^+ = -\bar{\beta}_i \geq \bar{\alpha}_{j^*} + \epsilon \geq \bar{\alpha}_j^+$, where the first inequality follows from the definition of ϵ. In addition, by construction $-\bar{\beta}_i^- = -\bar{\beta}_i > \bar{\alpha}_{j^*} \geq \bar{\alpha}_j^-$ for all $j \in P_i$.

(b) If $-\bar{\beta}_i < \bar{\alpha}_{j^*}$, then $j \notin \mathcal{J}(j^*)$ for all $j \in P_i$ because otherwise $-\bar{\beta}_i \geq \bar{\alpha}_j = \bar{\alpha}_{j^*}$. It follows that $\bar{\alpha}_j^+ = \bar{\alpha}_j^- = \bar{\alpha}_j$ and so $-\bar{\beta}_i^+ = -\bar{\beta}_i \geq \bar{\alpha}_j = \bar{\alpha}_j^+$ and $-\bar{\beta}_i^- = -\bar{\beta}_i \geq \bar{\alpha}_j = \bar{\alpha}_j^-$. □

Appendix E Proof of Theorem 1

Proof: First, pick any $(\alpha, \beta) \in \Lambda$ that satisfies the optimality conditions (a)–(b) stated in Lemma 2. We shall show that there exists a feasible solution (t, s, r, p) to formulation (7a)–(7d) that attains the same objective function value as $F(\alpha, \beta)$. To this end, for all $j \in [J]$, we let $t_j = 1$ if $\alpha_j = c^{\mathrm{x}}$ and $t_j = 0$ if $\alpha_j = c^{\mathrm{e}}$. In addition, for all $i \in [I]$, if $\alpha_j = c^{\mathrm{e}}$ for all $j \in P_i$ then we let $s_{ij} = 0$ for all $j \in P_i$; and otherwise, we pick the largest $j^* \in P_i$ such that $\alpha_{j^*} = c^{\mathrm{x}}$, and let $s_{ij^*} = 1$ and all other $s_{ij} = 0$. Also, we define r and p as in (7c) and (7d), respectively. By construction (t, s, r, p) satisfies constraints (7b)–(7d). It follows that the objective function value of (t, s, r, p) equals

$$\begin{aligned}
&\sum_{j=1}^{J} \left(c_j^{\mathrm{t}} t_j + c_j^{\mathrm{r}} r_j\right) + \sum_{i=1}^{I} \Big(c_i^{\mathrm{p}} p_i + \sum_{j \in P_i} c_i^{\mathrm{s}} s_{ij}\Big) \\
= \; & \sum_{j:\alpha_j = c^{\mathrm{x}}} \left\{ \Big[(-c^{\mathrm{x}} - \gamma_j) w_j\Big]_+ + \sup_{\tilde{d}_j \in [d_j^{\mathrm{L}}, d_j^{\mathrm{U}}]_{\mathbb{Z}}} \Big\{ c^{\mathrm{x}} \tilde{d}_j - \sum_{q=1}^{Q} \rho_{jq} \tilde{d}_j^q \Big\} \right\} \\
& + \sum_{j:\alpha_j = c^{\mathrm{e}}} \left\{ \Big[(-c^{\mathrm{e}} - \gamma_j) w_j\Big]_+ + \sup_{\tilde{d}_j \in [d_j^{\mathrm{L}}, d_j^{\mathrm{U}}]_{\mathbb{Z}}} \Big\{ c^{\mathrm{e}} \tilde{d}_j - \sum_{q=1}^{Q} \rho_{jq} \tilde{d}_j^q \Big\} \right\}
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{i=1}^{I}\left\{\mathbb{1}_{\{c^{\mathrm{x}}\}}\Big(\max_{j\in P_i}\{\alpha_j\}\Big)\Big[(-c^{\mathrm{x}}-\lambda_i)y_i\Big]_+ + \mathbb{1}_{\{c^{\mathrm{e}}\}}\Big(\max_{j\in P_i}\{\alpha_j\}\Big)\Big[(-c^{\mathrm{e}}-\lambda_i)y_i\Big]_+\right\}\\
=\;& \sum_{j=1}^{J}\left\{\Big[(-\alpha_j-\gamma_j)w_j\Big]_+ + \sup_{\tilde d_j\in[d_j^{\mathrm{L}},d_j^{\mathrm{U}}]_{\mathbb{Z}}}\Big\{\alpha_j\tilde d_j-\sum_{q=1}^{Q}\rho_{jq}\tilde d_j^q\Big\}\right\}+\sum_{i=1}^{I}\Big[(\beta_i-\lambda_i)y_i\Big]_+\\
=\;& F(\alpha,\beta),
\end{aligned}
$$

where the first equality follows from the definition of (t, s, r, p) and the second equality follows from the optimality condition (b).

Second, pick any feasible solution (t, s, r, p) to formulation (7a)–(7d). We construct an $(\alpha, \beta) \in \Lambda$ such that it satisfies the optimality conditions (a)–(b) and $F(\alpha, \beta)$ equals the objective function value (7a) of (t, s, r, p). Specifically, for all $j \in [J]$, we let $\alpha_j = c^{\mathrm{x}} t_j + c^{\mathrm{e}} r_j$ and, for all $i \in [I]$, $\beta_i = -c^{\mathrm{e}} p_i - c^{\mathrm{x}} \sum_{j\in P_i} s_{ij}$. Then, for all $i \in [I]$ and $j \in P_i$,

$$
\begin{aligned}
\beta_i+\alpha_j \;&=\; -c^{\mathrm{e}}p_i - c^{\mathrm{x}}\sum_{j\in P_i}s_{ij}+c^{\mathrm{x}}t_j+c^{\mathrm{e}}r_j\\
&=\; -c^{\mathrm{e}}\Big(1-\sum_{j\in P_i}s_{ij}\Big)-c^{\mathrm{x}}\sum_{j\in P_i}s_{ij}+c^{\mathrm{x}}t_j+c^{\mathrm{e}}(1-t_j)\\
&= (c^{\mathrm{e}}-c^{\mathrm{x}})\Big(\sum_{j\in P_i}s_{ij}-t_j\Big)\;\le\; 0,
\end{aligned}
$$

where the first equality is due to constraints (7c)–(7d) and the inequality is due to constraints (6d). Next, we have $\alpha_j \in \{c^{\mathrm{x}}, c^{\mathrm{e}}\}$ for all $j \in [J]$ due to constraint (7c). Hence, $(\alpha, \beta) \in \Lambda$. Also, for all $i \in [I]$, if $\sum_{\ell\in P_i} s_{i\ell} = 0$ then $p_i = 1$ due to constraints (7d) and $t_j = 0$ for all $j \in P_i$ due to constraint (6d). It follows that $\alpha_j = c^{\mathrm{e}}$ for all $j \in P_i$ and so $\beta_i = -\max\{\alpha_j : j \in P_i\}$. On the other hand, if $\sum_{\ell\in P_i} s_{i\ell} = 1$ then $p_i = 0$ and there exists an $j^* \in P_i$ with $t_{j^*} = 1$ due to constraints (6b). It follows that $\alpha_{j^*} = c^{\mathrm{x}}$ and so $\beta_i = -\max\{\alpha_j : j \in P_i\}$. Hence, (α, β) satisfies the optimality conditions (a)–(b). Finally,

$$
\begin{aligned}
F(\alpha,\beta) =\;& \sum_{j=1}^{J}\left\{\Big[(-c^{\mathrm{x}}t_j-c^{\mathrm{e}}r_j-\gamma_j)w_j\Big]_+ + \sup_{\tilde d_j\in[d_j^{\mathrm{L}},d_j^{\mathrm{U}}]_{\mathbb{Z}}}\Big\{\big(c^{\mathrm{x}}t_j+c^{\mathrm{e}}r_j\big)\tilde d_j-\sum_{q=1}^{Q}\rho_{jq}\tilde d_j^q\Big\}\right\}\\
&+\sum_{i=1}^{I}\Big[\Big(-c^{\mathrm{e}}p_i-c^{\mathrm{x}}\sum_{j\in P_i}s_{ij}-\lambda_i\Big)y_i\Big]_+\\
=\;& \sum_{j=1}^{J}\big(c_j^{\mathrm{t}}t_j+c_j^{\mathrm{r}}r_j\big)+\sum_{i=1}^{I}\Big(c_i^{\mathrm{p}}p_i+\sum_{j\in P_i}c_i^{\mathrm{s}}s_{ij}\Big)
\end{aligned}
$$

by the definition of (α, β) and constraints (7c)–(7d). This completes the proof. □

Appendix F Proof of Proposition 2

Proof: Let $G(\gamma, \lambda, \rho)$ be the objective function of problem (5a)–(5b), $(\hat{\gamma}, \hat{\lambda}, \hat{\rho})$ be any feasible solution, and S^* be the set of optimal solution to problem $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$ for the given $(\hat{\gamma}, \hat{\lambda}, \hat{\rho})$. Suppose that there exists a $j \in [J]$ such that $\hat{\gamma}_j < -c^{\mathrm{x}}$. Then, $-\hat{\gamma}_j - \alpha_j^* > 0$ and $[(-\hat{\gamma}_j - \alpha_j^*)w_j]_+ = (-\hat{\gamma}_j - \alpha_j^*)w_j$ for all $(\alpha^*, \beta^*) \in S^*$ because $\alpha_j^* \leq c^{\mathrm{x}}$ by Lemma 2. Additionally, due to Lemma 2, we can replace polyhedron Λ by the (compact) set of its extreme points $\mathrm{ex}(\Lambda)$ without loss of optimality, i.e., $\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta) = \max_{(\alpha,\beta)\in\mathrm{ex}(\Lambda)} F(\alpha, \beta)$. It then follows from Theorem 2.87 in [35] that, for all subgradient $\varpi \in \partial G(\hat{\gamma}, \hat{\lambda}, \hat{\rho})$, the entry in ϖ with regard to variable γ_j at $(\hat{\gamma}, \hat{\lambda}, \hat{\rho})$ equals $f_j(w_j) - w_j$, i.e., $\varpi(\gamma_j)|_{(\hat{\gamma},\hat{\lambda},\hat{\rho})} = f_j(w_j) - w_j \leq 0$. Noting that $\varpi(\gamma_j)|_{(\hat{\gamma},\hat{\lambda},\hat{\rho})} \leq 0$ holds valid whenever $\hat{\gamma}_j < -c^{\mathrm{x}}$, we can increase $\hat{\gamma}_j$ to $-c^{\mathrm{x}}$ without any loss of optimality.

Now suppose that $\hat{\gamma}_j > 0$. Then, we have $-\hat{\gamma}_j - \alpha_j^* < 0$ and $[(-\hat{\gamma}_j - \alpha_j^*)w_j]_+ = 0$ for all $(\alpha^*, \beta^*) \in S^*$ because $\alpha^* \geq 0$ by Lemma 2. It follows from a similar implication as in the previous paragraph that, for all subgradient $\varpi \in \partial G(\hat{\gamma}, \hat{\lambda}, \hat{\rho})$, we have $\varpi(\gamma_j)|_{(\hat{\gamma},\hat{\lambda},\hat{\rho})} = f_j(w_j) \geq 0$. Noting that this holds valid whenever $\hat{\gamma}_j > 0$, we can decrease $\hat{\gamma}_j$ to 0 without any loss of optimality. Therefore, there exists an optimal solution $(\gamma^*, \lambda^*, \rho^*)$ to problem (5a)–(5b) such that $\gamma_j^* \in [-c^{\mathrm{x}}, 0]$ for all $j \in [J]$.

Following a similar proof, we can show that there exists an optimal solution $(\gamma^*, \lambda^*, \rho^*)$ to problem (5a)–(5b) such that $\lambda_i^* \in [-c^{\mathrm{x}}, 0]$ for all $i \in [I]$. We omit the details for the sake of saving space. □

Appendix G Proof of Theorem 3

Proof: In each iteration of Algorithm 1, we solve a relaxation of the (DRNS) reformulation (9a)–(9d). It follows that, if the algorithm stops in an iteration and returns a solution (u^*, v^*) then (u^*, v^*) satisfies all the constraints (9c) because of Step 5. Then, (u^*, v^*) is feasible to formulation (9a)–(9d) and meanwhile optimal to its relaxation. Hence, (u^*, v^*) is optimal to formulation (9a)–(9d), i.e., optimal to (DRNS).

It remains to show that Algorithm 1 stops in a finite number of iterations. To see this, we notice that the set $\mathcal{H}$ contains a finite number of elements. Indeed, binary variables t and s only have a finite number of possible values. Although r and p are continuous variables, they also only have a finite number of possible values due to constraints (7c)–(7d). □

Appendix H Proof of Lemma 3

Proof: Pick any $i \in [I]$ and $j \in P_i$. We note that $\sum_{\ell \in P_i} s_{i\ell} \in \{0,1\}$ due to constraints (6a) and discuss the following three cases. First, if $\sum_{\ell \in P_i} s_{i\ell} = 0$, i.e., if $s_{i\ell} = 0$ for all $\ell \in P_i$, then $t_j = 0$ by constraints (6d). The inequalities (10) hold valid because all $s_{i\ell} \geq 0$.

Second, if $\sum_{\ell \in P_i} s_{i\ell} = 1$ and $\sum_{\ell \in P_i : \ell \geq j} s_{ij} = 1$, then inequalities (10) hold valid because $t_j \leq 1$.

Third, if $\sum_{\ell \in P_i} s_{i\ell} = 1$ and $\sum_{\ell \in P_i : \ell \geq j} s_{ij} = 0$, then there exists some $k \in P_i$, $k < j$ such that $s_{ik=1}$. Then, $t_j \leq 1 - s_{ik} = 0$ by constraints (6c). Inequalities (10) follow and the proof is complete. □

Appendix I Proof of Theorem 4

Proof: As $I = 1$ and $P_1 = [J]$, we drop the index i and re-write $\overline{\mathcal{H}} = \big\{(t,s,r,p) : \ \sum_{j=1}^{J} s_j \leq 1, \ \sum_{j=1}^{J} s_j + p = 1, \ s_j \leq t_j, \ t_j \leq \sum_{\ell=j}^{J} s_\ell, \ t_j + r_j = 1, \ s_j, t_j \in \mathbb{R}_+, \ \forall j \in [J]\big\}$. To show that $\text{conv}(\mathcal{H}) = \overline{\mathcal{H}}$, we first note that $\mathcal{H} \subseteq \overline{\mathcal{H}}$. This is because inequalities (10) are satisfied by all $(t,s,r,p) \in \mathcal{H}$. It follows that $\mathcal{H} \subseteq \overline{\mathcal{H}}$ and hence $\text{conv}(\mathcal{H}) \subseteq \overline{\mathcal{H}}$.

Second, we prove that $\overline{\mathcal{H}} \subseteq \text{conv}(\mathcal{H})$. To this end, we claim that optimizing any linear objective function over $\overline{\mathcal{H}}$ yields at least an optimal solution that lies in $\mathcal{H}$ (see [31]). If this claim holds valid then all the extreme points of $\overline{\mathcal{H}}$ lie in $\mathcal{H}$ and hence $\overline{\mathcal{H}} \subseteq \text{conv}(\mathcal{H})$ by the Minkowski's theorem on polyhedron. To prove this claim, we consider a linear program

$$\max_{t,s,r,p \geq 0} \ \sum_{j=1}^{J} (a_j^{\text{s}} s_j + a_j^{\text{t}} t_j + a_j^{\text{r}} r_j + a^{\text{p}} p) \tag{24a}$$

$$\text{s.t.} \ \sum_{j=1}^{J} s_j \leq 1, \tag{24b}$$

$$s_j \leq t_j, \quad \forall j \in [J], \tag{24c}$$

$$t_j \leq \sum_{\ell=j}^{J} s_\ell, \quad \forall j \in [J], \tag{24d}$$

$$t_j + r_j = 1, \quad \forall j \in [J], \tag{24e}$$

$$\sum_{j=1}^{J} s_j + p = 1, \tag{24f}$$

where a_j^{s}, a_j^{t}, a_j^{r}, and a^{p} represent arbitrary objective function coefficients. By the last two constraints, we re-write this linear program as

$$\max_{t,s \geq 0} \ \sum_{j=1}^{J} \big[(a_j^{\text{s}} - a^{\text{p}}) s_j + (a_j^{\text{t}} - a_j^{\text{r}}) t_j\big] + a^{\text{p}} + \sum_{j=1}^{J} a_j^{\text{r}}$$

$$
\begin{aligned}
&\text{s.t.}\quad \text{(24b)–(24d)}\\
= &\max_{s\geq 0:\ \text{(24b)}} \sum_{j=1}^{J}(a^{\mathrm{s}}_j - a^{\mathrm{p}})s_j + \max_{t\geq 0:\ \text{(24c)–(24d)}} \sum_{j=1}^{J}(a^{\mathrm{t}}_j - a^{\mathrm{r}}_j)t_j + a^{\mathrm{p}} + \sum_{j=1}^{J} a^{\mathrm{r}}_j\\
= &\max_{s\geq 0:\ \text{(24b)}} \sum_{j=1}^{J}(a^{\mathrm{s}}_j - a^{\mathrm{p}})s_j + \sum_{j=1}^{J} \max_{s_j\leq t_j\leq \sum_{\ell=j}^{J} s_\ell}(a^{\mathrm{t}}_j - a^{\mathrm{r}}_j)t_j + a^{\mathrm{p}} + \sum_{j=1}^{J} a^{\mathrm{r}}_j \qquad \text{(24g)}\\
= &\max_{s\geq 0:\ \text{(24b)}} \sum_{j=1}^{J}(a^{\mathrm{s}}_j - a^{\mathrm{p}})s_j + \sum_{j:a^{\mathrm{t}}_j\geq a^{\mathrm{r}}_j}(a^{\mathrm{t}}_j - a^{\mathrm{r}}_j)\sum_{\ell=j}^{J} s_\ell + \sum_{j:a^{\mathrm{t}}_j< a^{\mathrm{r}}_j}(a^{\mathrm{t}}_j - a^{\mathrm{r}}_j)s_j + a^{\mathrm{p}} + \sum_{j=1}^{J} a^{\mathrm{r}}_j\\
= &\max_{s\geq 0:\ \text{(24b)}} \sum_{j=1}^{J}\Big(a^{\mathrm{s}}_j - a^{\mathrm{p}} + a^{\mathrm{t}}_j - a^{\mathrm{r}}_j + \sum_{\ell=1}^{j-1}(a^{\mathrm{t}}_j - a^{\mathrm{r}}_j)_+\Big)s_j + a^{\mathrm{p}} + \sum_{j=1}^{J} a^{\mathrm{r}}_j. \qquad \text{(24h)}
\end{aligned}
$$

As the formulation (24h) optimizes a linear function of s over a simplex, there exists an optimal solution s^* to (24h), and hence an optimal solution (t^*, s^*, r^*, p^*) to (24a)–(24f), such that $s^* \in \{0\} \cup \{e_j : j \in [J]\}$ and, for all $t \in [J]$, either $t^*_j = s^*_j$ or $t^*_j = \sum_{\ell=j}^{J} s^*_\ell$ in view of the inner maximization problem in (24g). It follows that $(t^*, s^*) \in \mathbb{B}^{2J}$ and hence $(t^*, s^*, r^*, p^*) \in \mathcal{H}$. This proves that $\mathrm{conv}(\mathcal{H}) = \overline{\mathcal{H}}$.

Third, it follows from the above convex hull result and Theorem 1 that $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta) = \max_{(t,s,r,p)\in\overline{\mathcal{H}}}\big\{\sum_{j=1}^{J}(c^{\mathrm{t}}_j t_j + c^{\mathrm{r}}_j r_j) + c^{\mathrm{p}}_1 p_1\big\}$ (note that $c^{\mathrm{s}}_1 = 0$ by (9f)). Then, following (24h) we have

$$
\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta) \;=\; \max_{s\geq 0:\ \sum_{j=1}^{J} s_{1j}\leq 1} \sum_{j=1}^{J}\Big(-c^{\mathrm{p}}_1 + c^{\mathrm{t}}_j - c^{\mathrm{r}}_j + \sum_{\ell=1}^{j-1}(c^{\mathrm{t}}_j - c^{\mathrm{r}}_j)_+\Big)s_{1j} + c^{\mathrm{p}}_1 + \sum_{j=1}^{J} c^{\mathrm{r}}_j,
$$

which optimizes a linear function over the simplex $\{s \geq 0 :\ \sum_{j=1}^{J} s_{1j} \leq 1\}$. Enumerating the extreme points of this simplex, i.e., $\{0\}\cup\{s_{1j} = 1, s_{1\ell} = 0,\ \forall \ell \neq j\}_{j=1}^{J}$, yields the claimed closed-form solution of $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$. This completes the proof. □

Appendix J Proof of Proposition 3

Proof: By Theorem 4, constraints (9c) are equivalent to

$$
\theta \;\geq\; \max\Bigg\{c^{\mathrm{p}}_1 + \sum_{\ell=1}^{J} c^{\mathrm{r}}_\ell,\ \max_{j\in[J]}\Big\{c^{\mathrm{t}}_j + \sum_{\ell=1}^{j-1}\max\{c^{\mathrm{t}}_\ell, c^{\mathrm{r}}_\ell\} + \sum_{\ell=j+1}^{J} c^{\mathrm{r}}_\ell\Big\}\Bigg\},
$$

where c^{t}_j, c^{r}_j, and c^{p}_1 are computed by (9e)–(9h). Defining auxiliary variables

$$
\left.\begin{aligned}
\zeta_j &:= \Big[(-c^{\mathrm{e}} - \gamma_j)w^{\mathrm{L}}_j - \textstyle\sum_{k=1}^{w^{\mathrm{U}}_j - w^{\mathrm{L}}_j}(\varphi_{jk} + c^{\mathrm{e}} u_{jk})\Big]_+\\
\eta^{\mathrm{x}}_j &:= \sup_{\tilde{d}_j\in[d^{\mathrm{L}}_j, d^{\mathrm{U}}_j]_{\mathbb{Z}}}\big\{c^{\mathrm{x}}\tilde{d}_j - \textstyle\sum_{q=1}^{Q}\tilde{d}^q_j\rho_{jq}\big\}\\
\eta^{\mathrm{e}}_j &:= \sup_{\tilde{d}_j\in[d^{\mathrm{L}}_j, d^{\mathrm{U}}_j]_{\mathbb{Z}}}\big\{c^{\mathrm{e}}\tilde{d}_j - \textstyle\sum_{q=1}^{Q}\tilde{d}^q_j\rho_{jq}\big\}
\end{aligned}\right\}\quad \forall j\in[J],
$$

$$\text{and} \quad \phi_1 \ := \ \Big[(-c^{\mathrm{e}} - \lambda_1) y_1^{\mathrm{L}} - \sum_{\ell=1}^{y_1^{\mathrm{U}} - y_1^{\mathrm{L}}} (\nu_{1\ell} + c^{\mathrm{e}} v_{1\ell})\Big]_+,$$

we have $c_j^{\mathrm{t}} = \eta_j^{\mathrm{x}}$, $c_j^{\mathrm{r}} = \zeta_j + \eta_j^{\mathrm{e}}$ for all $j \in [J]$, and $c_1^{\mathrm{p}} = \phi_1$. It follows that constraints (9c) are equivalent to

$$\theta \ \geq \ \max\Bigg\{\phi_1 + \sum_{\ell=1}^{J}(\zeta_\ell + \eta_\ell^{\mathrm{e}}),\ \max_{j \in [J]}\Big\{\eta_j^{\mathrm{x}} + \sum_{\ell=1}^{j-1} \max\{\eta_\ell^{\mathrm{x}}, \zeta_\ell + \eta_\ell^{\mathrm{e}}\} + \sum_{\ell=j+1}^{J} (\zeta_\ell + \eta_\ell^{\mathrm{e}})\Big\}\Bigg\}$$

$$\Leftrightarrow \ \left\{ \begin{array}{l} \theta \ \geq \ \phi_1 + \sum_{\ell=1}^{J}(\zeta_\ell + \eta_\ell^{\mathrm{e}}) \\ \theta \ \geq \ \eta_j^{\mathrm{x}} + \sum_{\ell=1}^{j-1} \max\{\eta_\ell^{\mathrm{x}}, \zeta_\ell + \eta_\ell^{\mathrm{e}}\} + \sum_{\ell=j+1}^{J}(\zeta_\ell + \eta_\ell^{\mathrm{e}}), \quad \forall j \in [J] \end{array} \right.$$

$$\Leftrightarrow \ \exists \{\chi_j\}_{j=1}^{J} : \ \left\{ \begin{array}{l} \theta \ \geq \ \phi_1 + \sum_{\ell=1}^{J}(\zeta_\ell + \eta_\ell^{\mathrm{e}}) \\ \theta \ \geq \ \eta_j^{\mathrm{x}} + \sum_{\ell=1}^{j-1} \chi_\ell + \sum_{\ell=j+1}^{J}(\zeta_\ell + \eta_\ell^{\mathrm{e}}), \quad \forall j \in [J] \\ \chi_j \geq \eta_j^{\mathrm{x}}, \quad \chi_j \geq \zeta_j + \eta_j^{\mathrm{e}}, \quad \forall j \in [J] \end{array} \right. \tag{25}$$

Replacing constraints (9c) with (25) in the formulation (9a)–(9d) and incorporating the definition of the auxiliary variables ζ_j, η_j^{x}, η_j^{e}, and ϕ_1 leads to the claimed reformulation of (DRNS). This completes the proof. □

Appendix K Proof of Proposition 4

We start by proving the following technical lemma.

Lemma 4 *Consider sets $A_i \subseteq \mathbb{R}^{k_i}$ for all $i \in [I]$ and let $A := \Pi_{i=1}^{I} A_i$. Then, conv$(A) = \Pi_{i=1}^{I}$conv(A_i).*

Proof: First, as $A = \Pi_{i=1}^{I} A_i$ and $A_i \subseteq \mathrm{conv}(A_i)$, we have $A \subseteq \Pi_{i=1}^{I}\mathrm{conv}(A_i)$ and hence $\mathrm{conv}(A) \subseteq \Pi_{i=1}^{I}\mathrm{conv}(A_i)$. Second, to show that $\Pi_{i=1}^{I}\mathrm{conv}(A_i) \subseteq \mathrm{conv}(A)$, we pick any $a \in \Pi_{i=1}^{I}\mathrm{conv}(A_i)$ and prove that $a \in \mathrm{conv}(A)$. To this end, we denote $a := [a_1, \ldots, a_I]^\top$, where $a_i \in \mathrm{conv}(A_i)$ for all $i \in [I]$. Then, for all $i \in [I]$, there exist $\{\lambda_i^{n_i}\}_{n_i=1}^{N_i}$ and $\{a_i^{n_i}\}_{n_i=1}^{N_i}$ such that each $\lambda_i^{n_i} \geq 0$, each $a_i^{n_i} \in A_i$, $\sum_{n_i=1}^{N_i} \lambda_i^{n_i} = 1$, and $\sum_{n_i=1}^{N_i} \lambda_i^{n_i} a_i^{n_i} = a_i$ for all $i \in [I]$.

Denote set $\mathcal{N} := \{(n_1, \ldots, n_I) : n_i \in [N_i],\ \forall i \in [I]\}$, vector $a^n := [a_1^{n_1}, \ldots, a_I^{n_I}]^\top$ for all $n := [n_1, \ldots, n_I]^\top \in \mathcal{N}$, and scalar $\lambda^n := \Pi_{i=1}^{I} \lambda_i^{n_i}$ for all $n \in \mathcal{N}$. Then, $\lambda^n \geq 0$ and $a^n \in A$ for all $n \in \mathcal{N}$. In addition, $\sum_{n \in \mathcal{N}} \lambda^n = \sum_{n \in \mathcal{N}} \Pi_{i=1}^{I} \lambda_i^{n_i} = (\lambda_1^1 + \cdots + \lambda_1^{N_1})(\lambda_2^1 + \cdots + \lambda_2^{N_2}) \cdots (\lambda_I^1 + \cdots + \lambda_I^{N_I}) = 1$. Furthermore, for all $i \in [I]$, we have

$$\begin{aligned} \sum_{n \in \mathcal{N}} \lambda^n a_i^{n_i} \ &= \ \sum_{m_i=1}^{N_i} \sum_{n \in \mathcal{N} : n_i = m_i} \lambda^n a_i^{n_i} \\ &= \ \sum_{m_i=1}^{N_i} a_i^{m_i} \sum_{n \in \mathcal{N} : n_i = m_i} \lambda^n \end{aligned}$$

$$= \sum_{m_i=1}^{N_i} a_i^{m_i} \lambda_i^{m_i}$$
$$= a_i,$$

where third equality is because, for fixed $i \in [I]$ and $m_i \in [N_i]$, $\sum_{n\in\mathcal{N}:n_i=m_i} \lambda^n = (\lambda_1^1 + \cdots + \lambda_1^{N_1}) \cdots (\lambda_{m_i-1}^1 + \cdots + \lambda_{m_i-1}^{N_{m_i-1}}) \lambda_i^{m_i} (\lambda_{m_i+1}^1 + \cdots + \lambda_{m_i+1}^{N_{m_i+1}}) \cdots (\lambda_I^1 + \cdots + \lambda_I^{N_I}) = \lambda_i^{m_i}$, and the last inequality follows from the definitions of $\{\lambda_i^{m_i}\}_{m_i=1}^{N_i}$ and $\{a_i^{m_i}\}_{m_i=1}^{N_i}$. It follows that $a \equiv [a_1, \ldots, a_I]^\top = \sum_{n\in\mathcal{N}} \lambda^n [a_1^{n_1}, \ldots, a_I^{n_I}]^\top \equiv \sum_{n\in\mathcal{N}} \lambda^n a^n$ and hence $a \in \text{conv}(A)$. This completes the proof. □

We are now ready to present the main proof of this section.

Proof of Proposition 4: First, as $\mathcal{H}$ is separable in index i, we have $\mathcal{H} = \Pi_{i=1}^I \mathcal{H}_i$, where each $\mathcal{H}_i$ is defined as

$$\mathcal{H}_i := \Big\{(t, s_i, r, p_i) : \sum_{j\in P_i} s_{ij} \le 1, \tag{26a}$$
$$s_{ij} \le t_j, \quad \forall j \in P_i, \tag{26b}$$
$$t_j + s_{i\ell} \le 1, \quad \forall j, \ell \in P_i : j > \ell, \tag{26c}$$
$$t_j \le \sum_{\ell\in P_i} s_{i\ell}, \quad \forall j \in P_i, \tag{26d}$$
$$t_j, s_{ij} \in \mathbb{B}, \quad \forall j \in P_i, \tag{26e}$$
$$t_j + r_j = 1, \quad \forall j \in P_i, \tag{26f}$$
$$p_i + \sum_{j\in P_i} s_{ij} = 1\Big\}. \tag{26g}$$

Following a similar proof as that of Theorem 4, we can show that incorporating inequalities (10) produces the convex hull of $\mathcal{H}_i$, i.e., $\text{conv}(\mathcal{H}_i) = \{(t, s, r, p) \ge 0 : \text{(26a)–(26b), (26f)–(26g)}, \ t_j \le \sum_{\ell\in P_i:\ell\ge j} s_{ij}, \ \forall j \in P_i\}$. Then, it follows from Lemma 4 that $\text{conv}(\mathcal{H}) = \Pi_{i=1}^I \text{conv}(\mathcal{H}_i) = \big\{(t, s, r, p) \ge 0 : \text{(6a)–(6b), (7c)–(7d), (10)}\big\}$, as claimed.

Second, using this convex hull result, we have

$$\max_{(\alpha,\beta)\in\Lambda} F(\alpha, \beta)$$
$$= \max_{(t,s,r,p)\ge 0} \ \sum_{j=1}^J \big(c_j^{\text{t}} t_j + c_j^{\text{r}} r_j\big) + \sum_{i=1}^I c_i^{\text{p}} p_i$$
$$\text{s.t.} \quad \text{(6a)–(6b), (7c)–(7d), (10)}$$
$$= \sum_{i=1}^I \Big\{ c_i^{\text{p}} + \sum_{j\in P_i} c_j^{\text{r}} + \max_{(t,s)\ge 0:\text{(6a)–(6b), (10)}} \sum_{j\in P_i} \big[(c_j^{\text{t}} - c_j^{\text{r}}) t_j - c_i^{\text{p}} s_{ij}\big] \Big\}$$
$$= \sum_{i=1}^I \Big\{ c_i^{\text{p}} + \sum_{j\in P_i} c_j^{\text{r}} + \max_{s\ge 0:\text{(6a)}} \sum_{j\in P_i} \Big[-c_i^{\text{p}} + c_j^{\text{t}} - c_j^{\text{r}} + \sum_{\ell\in P_i:\ell<j} (c_j^{\text{t}} - c_j^{\text{r}})_+\Big] s_{ij} \Big\}$$

$$= \sum_{i=1}^{I}\Big\{c_i^{\mathrm{p}} + \sum_{j\in P_i} c_j^{\mathrm{r}} + \max\Big\{0,\ \max_{j\in P_i}\big\{-c_i^{\mathrm{p}} + c_j^{\mathrm{t}} - c_j^{\mathrm{r}} + \sum_{\ell\in P_i:\ell<j}(c_j^{\mathrm{t}} - c_j^{\mathrm{r}})_+\big\}\Big\}\Big\}$$

$$= \sum_{i\in[I]} \max\left\{c_i^{\mathrm{p}} + \sum_{\ell\in P_i} c_\ell^{\mathrm{r}},\ \max_{j\in P_i}\Big\{c_j^{\mathrm{t}} + \sum_{\ell\in P_i:\ell<j}\max\{c_\ell^{\mathrm{t}}, c_\ell^{\mathrm{r}}\} + \sum_{\ell\in P_i:\ell>j} c_\ell^{\mathrm{r}}\Big\}\right\}.$$

Third, constraints (9c) in the reformulation of (DRNS) are equivalent to $\theta \geq \max_{i\in[I]}\{\theta_i\}$, where

$$\theta_i \ \geq \ \max\left\{c_i^{\mathrm{p}} + \sum_{\ell\in P_i} c_\ell^{\mathrm{r}},\ \max_{j\in P_i}\Big\{c_j^{\mathrm{t}} + \sum_{\ell\in P_i:\ell<j}\max\{c_\ell^{\mathrm{t}}, c_\ell^{\mathrm{r}}\} + \sum_{\ell\in P_i:\ell>j} c_\ell^{\mathrm{r}}\Big\}\right\}.$$

The claimed reformulation of (DRNS) then follows from a similar proof as that of Proposition 3. □

Appendix L Proof of Theorem 5

Proof: First, by construction and Theorem 1, the DP yields the same optimal value as $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$. Second, each trajectory of states $\widehat{t}_1, (\widehat{t}_1, \widehat{t}_2)\ldots, (\widehat{t}_1, \widehat{t}_I)$ in the DP corresponds to a S-T path in the network $(\mathcal{N}, \mathcal{A})$, where the objective function value of the trajectory $V_I(\widehat{t}_1, \widehat{t}_I) + c_I^{\mathrm{p}}(1-\widehat{t}_I)(1-\widehat{t}_1)$ equals the length of the S-T path by definition of the arc lengths $c_{[m,n]}$. Likewise, each S-T path in $(\mathcal{N}, \mathcal{A})$ corresponds to a trajectory of states in the DP and the length of the path equals the objective function value of the trajectory. This proves that the length of the longest path in $(\mathcal{N}, \mathcal{A})$ equals $\max_{(\alpha,\beta)\in\Lambda} F(\alpha,\beta)$ and completes the proof. □

Appendix M Proof of Proposition 5

Proof: Taking the dual of the longest-path formulation yields

$$\begin{aligned}
\min_{\pi}\ \ & \pi_{\mathtt{S}} - \pi_{\mathtt{T}} \\
\text{s.t.}\ \ & \pi_{\mathtt{S}} - \pi_{\widehat{t}_1} \geq c_1^{\mathrm{t}}\widehat{t}_1 + c_1^{\mathrm{r}}(1-\widehat{t}_1), \quad \forall \widehat{t}_1 \in \mathbb{B}, \\
& \pi_{(\widehat{t}_1, \widehat{t}_{i-1})} - \pi_{(\widehat{t}_1, \widehat{t}_i)} \geq c_i^{\mathrm{t}} + (c_i^{\mathrm{r}} - c_i^{\mathrm{t}})(1-\widehat{t}_i) + c_{i-1}^{\mathrm{p}}(1-\widehat{t}_{i-1})(1-\widehat{t}_i) \\
& \forall \widehat{t}_{i-1}, \widehat{t}_i \in \mathbb{B},\ \forall i \in [2, I]_{\mathbb{Z}}, \\
& \pi_{(\widehat{t}_1, \widehat{t}_I)} - \pi_{\mathtt{T}} \geq c_I^{\mathrm{p}}(1-\widehat{t}_I)(1-\widehat{t}_1), \quad \forall \widehat{t}_1, \widehat{t}_I \in \mathbb{B},
\end{aligned}$$

where dual variables π are associated with the (primal) flow balance constraints and all dual constraints are associated with the primal variables x. The strong duality holds valid because the longest-path formulation is finitely optimal. The claimed reformulation of (DRNS) then follows from a similar proof as that of Proposition 3. □

Appendix N Proof of Proposition 6

Proof: We linearize the bilinear terms in constraints (17c)–(17e). First, for constraints (17d)–(17e), we define auxiliary variables

$$\zeta_{ij} := \zeta_j a_{ij}, \quad \eta^{\mathrm{e}}_{ij} := \eta^{\mathrm{e}}_j a_{ij}, \quad \text{and} \quad \eta^{\mathrm{x}}_{ij} := \eta^{\mathrm{x}}_j a_{ij}, \quad \forall i \in [I],\ \forall j \in [J]. \tag{27a}$$

We equivalently linearize these bilinear equalities as

$$\zeta_j = \sum_{i=1}^{I+1} \zeta_{ij}, \quad \eta^{\mathrm{x}}_j = \sum_{i=1}^{I+1} \eta^{\mathrm{x}}_{ij}, \quad \eta^{\mathrm{e}}_j = \sum_{i=1}^{I+1} \eta^{\mathrm{e}}_{ij}, \quad \forall j \in [J], \tag{27b}$$

$$0 \le \zeta_{ij} \le K a_{ij}, \quad -K a_{ij} \le \eta^{\mathrm{x}}_{ij} \le K a_{ij}, \quad -K a_{ij} \le \eta^{\mathrm{e}}_{ij} \le K a_{ij}. \tag{27c}$$

To see the equivalence, on the one hand, we notice that constraints (27b) follow from (27a) and (16a). Similarly, constraints (27c) follow from (27a) and the facts that a_{ij} are binary and $\zeta_j \ge 0$. On the other hand, constraints (27b) and (16a) imply that $\zeta_{ij} = \zeta_j$ if $a_{ij} = 1$, and constraints (27c) imply that $\zeta_{ij} = 0$ if $a_{ij} = 0$. We hence have $\zeta_{ij} = \zeta_j a_{ij}$. Likewise, we establish $\eta^{\mathrm{e}}_{ij} = \eta^{\mathrm{e}}_j a_{ij}$, $\eta^{\mathrm{x}}_{ij} = \eta^{\mathrm{x}}_j a_{ij}$, and hence constraints (27a). It follows that constraints (17d)–(17e) can be recast as $\theta_i \ge \eta^{\mathrm{x}}_{ij} + \sum_{\ell=1}^{j-1} \max\{\zeta_{i\ell} + \eta^{\mathrm{e}}_{i\ell}, \eta^{\mathrm{x}}_{i\ell}\} + \sum_{\ell=j+1}^{J} (\zeta_{i\ell} + \eta^{\mathrm{e}}_{i\ell})$, $\theta_i \ge \phi_i + \sum_{\ell=1}^{J} (\zeta_{i\ell} + \eta^{\mathrm{e}}_{i\ell})$, plus (27b)–(27c).

Second, we linearize constraint (17c) by claiming that

$$\begin{aligned}
&\Big(\theta_i + c^{\mathrm{y}} y^{\mathrm{L}}_i + g_i(y^{\mathrm{L}}_i)\lambda_i + \sum_{\ell=1}^{y^{\mathrm{U}}_i - y^{\mathrm{L}}_i} \big(\delta_{i\ell}\nu_{i\ell} + c^{\mathrm{y}} v_{i\ell}\big)\Big) o_i \\
=\ & \theta_i + c^{\mathrm{y}} y^{\mathrm{L}}_i o_i + g_i(y^{\mathrm{L}}_i)\lambda_i + \sum_{\ell=1}^{y^{\mathrm{U}}_i - y^{\mathrm{L}}_i} \big(\delta_{i\ell}\nu_{i\ell} + c^{\mathrm{y}} v_{i\ell}\big),
\end{aligned} \tag{27d}$$

which holds valid if $\theta_i = 0$, $\lambda_i = 0$, and $v_{i\ell} = 0$ for all $\ell \in [y^{\mathrm{U}}_i - y^{\mathrm{L}}_i]$ whenever $o_i = 0$ (note that variables $\nu_{i\ell}$ also vanish in this case because $\nu_{i\ell} = \lambda_i v_{i\ell}$). To this end, we incorporate constraints

$$-c^{\mathrm{x}} o_i \le \lambda_i \le 0 \tag{27e}$$

because $\lambda_i \in [-c^{\mathrm{x}}, 0]$ without loss of optimality by Proposition 2. This guarantees that $o_i = 0$ implies $\lambda_i = 0$. Additionally, we incorporate constraints

$$v_{i1} \le o_i. \tag{27f}$$

Then, $o_i = 0$ implies $v_{i1} = 0$ and hence $v_{i\ell} = 0$ for all $\ell \in [y^{\mathrm{U}}_i - y^{\mathrm{L}}_i]$ due to constraints (8f). Furthermore, $o_i = 0$ implies that $a_{ij} = 0$ for all $j \in [J]$ by constraints (16b). It follows that

$\zeta_{ij} = \eta^{\mathrm{e}}_{ij} = \eta^{\mathrm{x}}_{ij} = 0$ for all $j \in [J]$. It remains to ensure that $\phi_i = 0$ whenever $o_i = 0$. To that end, we replace constraints (12a) with

$$\phi_i \geq -c^{\mathrm{e}} y_i^{\mathrm{L}} o_i - y_i^{\mathrm{L}} \lambda_i - \sum_{\ell=1}^{y_i^{\mathrm{U}} - y_i^{\mathrm{L}}} (\nu_{i\ell} + c^{\mathrm{e}} v_{i\ell}). \tag{27g}$$

Indeed, if $o_i = 0$ then $\lambda_i = 0$ by constraints (27e) and $\nu_{i\ell} = v_{i\ell} = 0$ for all $\ell \in [y_i^{\mathrm{U}} - y_i^{\mathrm{L}}]$ by constraints (27f). Therefore, constraint (17c) is equivalently linearized through equality (27d) and incorporating constraints (27e)–(27g). This completes the proof. □

Appendix O Symmetry Breaking Inequalities for the (OPD) Model

We consider two types of symmetry among integer solutions. First, suppose that there are 2 pools and 4 units. The following two unit assignments lead to symmetric integer solutions: (i) assigning all units to pool 1 and no unit to pool 2 (i.e., $a_{1j} = 1$ and $a_{2j} = 0$ for all $j \in [4]$) and (ii) assigning all units to pool 2 and no unit to pool 1 (i.e., $a_{1j} = 0$ and $a_{2j} = 1$ for all $j \in [4]$). We call this "pool symmetry." To break this symmetry, we designate that all open pools have smaller indices than the closed ones. This designation breaks the pool symmetry because the above case (ii) is now prohibited. Accordingly, we add the following inequalities to the (OPD) formulation:

$$o_i \geq o_{i+1}, \quad \forall i \in [I-1].$$

Second, the following two unit assignments also lead to symmetric integer solutions: (iii) assigning units 1 and 3 to pool 1 and units 2 and 4 to pool 2 (i.e., $a_{11} = 1 - a_{12} = a_{13} = 1 - a_{14} = 1$ and $1 - a_{21} = a_{22} = 1 - a_{23} = a_{24} = 1$) and (iv) assigning units 1 and 3 to pool 2 and units 2 and 4 to pool 1 (i.e., $1 - a_{11} = a_{12} = 1 - a_{13} = a_{14} = 1$ and $a_{21} = 1 - a_{22} = a_{23} = 1 - a_{24} = 1$). We call this "unit symmetry." To break this symmetry, we rank the pools based on the smallest unit index in each pool. That is, we designate that the smallest unit index in pool i is smaller than that in pool $i+1$ for all $i \in [I-1]$, if both pools are opened. This designation breaks the unit symmetry because the above case (iv) is now prohibited. Accordingly, we add the following inequalities to the (OPD) formulation:

$$\sum_{\ell=1}^{j-1} a_{i\ell} \geq a_{(i+1)j}, \quad \forall i \in [I-1], \ \forall j \in [J].$$

Appendix P Input Parameters of the Instance Reported in Figure 5, Section 6.4

Table 4: Input parameters of the representative instance

unit j	unit 1	unit 2	unit 3	unit 4	unit 5	unit 6	unit 7
μ_{j1}	11.42	6.34	17.73	19.15	19.69	15.67	14.84
sd_j	5.05	4.03	6.44	17.06	16.39	16.52	15.92
A^{u}_j	0.97	0.98	0.98	0.61	0.75	0.67	0.67
A^{p}_i	0.99						
costs	$c^{\mathrm{w}} = 100$, $c^{\mathrm{y}} = 130$, $c^{\mathrm{x}} = 400$, $c^{\mathrm{e}} = 50$						

References

[1] James Ang, Fanwen Meng, and Jie Sun. Two-stage stochastic linear programs with incomplete information on uncertainty. *European Journal of Operational Research*, 233(1):16–22, 2014.

[2] Maya Bam, Zheng Zhang, Brian Denton, Mark P. Van Oyen, and Mary Duck. Planning models for skills-sensitive surgical nurse staffing. *Available Onlin: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2959005*, 2017.

[3] Jonathan F. Bard. Nurse scheduling models. *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

[4] Dimitris Bertsimas, Xuan Vinh Doan, Karthik Natarajan, and Chung-Piaw Teo. Models for minimax stochastic linear optimization problems with risk aversion. *Mathematics of Operations Research*, 35(3):580–602, 2010.

[5] Dimitris Bertsimas, Melvyn Sim, and Meilin Zhang. Adaptive distributionally robust optimization. *Management Science*, 65(2):604–618, 2018.

[6] Zoe Boutsioli. Demand variability, demand uncertainty and hospital costs: a selective survey of the empirical literature. *Global Journal of Health Science*, 2(1):138, 2010.

[7] MJ Brusco and MJ Showalter. Constrained nurse staffing analysis. *Omega*, 21(2):175–186, 1993.

[8] California Department of Health. California RN Staffing Ratio Law, 2004.

[9] Gerard M. Campbell. A two-stage stochastic program for scheduling and allocating cross-trained workers. *Journal of the Operational Research Society*, 62(6):1038–1047, 2011.

[10] Xi Chen, Tengyu Ma, Jiawei Zhang, and Yuan Zhou. Optimal design of process flexibility for general production systems. *Operations Research*, 67(2):516–531, 2019.

[11] Xi Chen, Jiawei Zhang, and Yuan Zhou. Optimal sparse designs for process flexibility via probabilistic expanders. *Operations Research*, 63(5):1159–1176, 2015.

[12] Ashley Davis, Sanjay Mehrotra, Jane Holl, and Mark S. Daskin. Nurse staffing under demand uncertainty to reduce costs and enhance patient safety. *Asia-Pacific Journal of Operational Research*, 31(01):1450005, 2014.

[13] Jitka Dupačová. Optimization under exogenous and endogenous uncertainty. *University of West Bohemia in Pilsen*, 2006.

[14] Fred F. Easton. Service completion estimates for cross-trained workforce schedules under uncertain attendance and demand. *Production and Operations Management*, 23(4):660–675, 2014.

[15] Martin Gaynor and Gerard F. Anderson. Uncertain demand, the structure of hospital costs, and the cost of empty hospital beds. *Journal of Health Economics*, 14(3):291–317, 1995.

[16] Linda V. Green, Sergei Savin, and Nicos Savva. "Nursevendor Problem": personnel staffing in the presence of endogenous absenteeism. *Management Science*, 59(10):2237–2256, 2013.

[17] Grani A. Hanasusanto and Daniel Kuhn. Conic programming reformulations of two-stage distributionally robust linear programs over wasserstein balls. *Operations Research*, 66(3):849–869, 2018.

[18] Peter JH Hulshof, Nikky Kortbeek, Richard J Boucherie, Erwin W Hans, and Piet JM Bakker. Taxonomic classification of planning decisions in health care: a structured review of the state of the art in OR/MS. *Health Systems*, 1(2):129–175, 2012.

[19] Robert R Inman, Dennis E Blumenfeld, and Arthur Ko. Cross-training hospital nurses to reduce staffing costs. *Health Care Management Review*, 30(2):116–125, 2005.

[20] William C Jordan and Stephen C Graves. Principles on the benefits of manufacturing process flexibility. *Management Science*, 41(4):577–594, 1995.

[21] Edward PC Kao and Maurice Queyranne. Budgeting costs of nursing in a hospital. *Management Science*, 31(5):608–621, 1985.

[22] Edward PC Kao and Grace G Tung. Aggregate nursing requirement planning in a public health care delivery system. *Socio-Economic Planning Sciences*, 15(3):119–127, 1981.

[23] Kibaek Kim and Sanjay Mehrotra. A two-stage stochastic integer programming approach to integrated staffing and scheduling with application to nurse management. *Operations Research*, 63(6):1431–1451, 2015.

[24] Qingxia Kong, Chung-Yee Lee, Chung-Piaw Teo, and Zhichao Zheng. Scheduling arrivals to a stochastic service delivery system using copositive cones. *Operations Research*, 61(3):711–726, 2013.

[25] Nikky Kortbeek, Aleida Braaksma, Christian AJ Burger, Piet JM Bakker, and Richard J Boucherie. Flexible nurse staffing based on hourly bed census predictions. *International Journal of Production Economics*, 161:167–180, 2015.

[26] Susan Feng Lu and Lauren Xiaoyuan Lu. Do mandatory overtime laws improve quality? staffing decisions and operational flexibility of nursing homes. *Management Science*, 63(11):3566–3585, 2016.

[27] Fengqiao Luo and Sanjay Mehrotra. Distributionally robust optimization with decision dependent ambiguity sets. *Available Online:* `https://arxiv.org/abs/1806.09215`, 2018.

[28] Kayse Lee Maass, Boying Liu, Mark S. Daskin, Mary Duck, Zhehui Wang, Rama Mwenesi, and Hannah Schapiro. Incorporating nurse absenteeism into staffing with demand uncertainty. *Health Care Management Science*, 20(1):141–155, 2017.

[29] Broos Maenhout and Mario Vanhoucke. An integrated nurse staffing and scheduling analysis for longer-term nursing staff allocation problems. *Omega*, 41(2):485–499, 2013.

[30] G. P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I–Convex underestimating problems. *Mathematical Programming*, 10(1):147–175, 1976.

[31] George L. Nemhauser and Laurence A. Wolsey. *Integer and Combinatorial Optimization*. John Wiley & Sons, 1999.

[32] Nilay Noyan, Gábor Rudolf, and Miguel Lejeune. Distributionally robust optimization with decision-dependent ambiguity set. *Available Online:* `http://www.optimization-online.org/DB_FILE/2018/09/6821.pdf`, 2018.

[33] Victoria Department of Health. Safe Patient Care (Nurse to Patient and Midwife to Patient Ratios) Act, 2015.

[34] Bureau of Labor Statistics. Absences from work of employed full-time wage and salary workers by occupation and industry, 2016.

[35] Andrzej P Ruszczyński. *Nonlinear optimization*, volume 13. Princeton University Press, 2006.

[36] Vincent W Slaugh, Alan A Scheller-Wolf, and Sridhar R Tayur. Consistent staffing for long-term care through on-call pools. *Production and Operations Management*, 2018.

[37] B. D. Stewart, D. B. Webster, S. Ahmad, and J. O. Matson. Mathematical models for developing a flexible workforce. *International Journal of Production Economics*, 36(3):243–254, 1994.

[38] Pamela Tevington. Mandatory nurse-patient ratios. *Medsurg Nursing*, 20(5):265, 2011.

[39] Vandankumar M. Trivedi. A mixed-integer goal programming model for nursing service budgeting. *Operations Research*, 29(5):1019–1034, 1981.

[40] Jorne Van den Bergh, Jeroen Beliën, Philippe De Bruecker, Erik Demeulemeester, and Liesje De Boeck. Personnel scheduling: A literature review. *European Journal of Operational Research*, 226(3):367–385, 2013.

[41] Wen-Ya Wang and Diwakar Gupta. Nurse absenteeism and staffing strategies for hospital inpatient units. *Manufacturing & Service Operations Management*, 16(3):439–454, 2014.

[42] Xuan Wang and Jiawei Zhang. Process flexibility: A distribution-free bound on the performance of k-chain. *Operations Research*, 63(3):555–571, 2015.

[43] John M Welton. Hospital nursing workforce costs, wages, occupational mix, and resource utilization. *Journal of Nursing Administration*, 41(7/8):309–314, 2011.

[44] Kum-Khiong Yang, Scott Webster, and Robert A. Ruben. An evaluation of worker cross training and flexible workdays in job shops. *IIE Transactions*, 39(7):735–746, 2007.