7-2018

On the design of sparse but efficient structures in operations

Zhenzhen YAN
National University of Singapore

Sarah Yini GAO
Singapore Management University, yngao@smu.edu.sg

Chung Piaw TEO
National University of Singapore

Submitted to *Management Science*
manuscript (Please, provide the mansucript number!)

On the Design of Sparse but Efficient Structures in Operations

Zhenzhen Yan
National University of Singapore, Singapore, yanzhenzhen8@gmail.com,

Sarah Yini Gao
Singapore Management University, Singapore, yngao@smu.edu.sg,

Chung Piaw Teo
National University of Singapore, Singapore, bizteocp@nus.edu.sg,

It is widely believed that "a little flexibility added at the right place can reap significant benefits for operations". Unfortunately, despite the extensive literature on this topic, we are not aware of any general methodology that can be used to guide managers to design sparse (i.e., slightly flexible) and yet efficient operations.

We address this issue using a distributionally robust approach to model the performance of a stochastic system under different process structures. We use the dual prices obtained from a related conic program to guide managers in the design process. This leads to a general solution methodology for the construction of efficient sparse structures for several classes of operational problems.

Our approach can be used to design simple yet efficient structures for workforce deployment and for any level of sparsity requirement, to respond to deviations and disruptions in the operational environment. Furthermore, in the case of the classical process flexibility problem, our methodology can recover the k-chain structures that are known to be extremely efficient for this type of problem when the system is balanced and symmetric. We can also obtain the analog of 2-chain for unbalanced system using this methodology.

Key words: Sparse and Efficient Operation, Sensitivity Analysis, Conic Program, Manufacturing Flexibility, Strong Duality

1. Introduction

There are two competing paradigms in the design of efficient operations: One argues for the standardization of business processes and practices to achieve operational excellence and cost leadership, and the other lauds the benefits of flexible operations to adapt and respond to changing business needs. Many operational design problems often reduce to a tussle between the two opposing paradigms—with more standardization you have less flexibility, and vice versa. Henry Ford, in describing his insistence on standardization in assembly-line operations, wrote in his autobiography that he had told his sales staff, facetiously, that "any customer can have a Model T painted in any color that he wants so long as it is black." In fact, Model Ts came in several colors; Ford made the comment to stress standardization's critical role.

Technological advancements in the field of automation and the Internet-of-Things, together with clever engineering concepts such as modular product design and synchronized engineering[1], have allowed some companies to reap the benefits of both worlds, to respond to swings in market sentiments and changing consumer demands at affordable cost. Honda Motor, for instance, "can seamlessly produce multiple models on a single assembly line, one after another, and switch over to a newly designed vehicle within hours. By contrast, it can take months for Honda's rivals to retool a factory for a new vehicle," according to Jeffrey Rothfeder (2014)[2].

This push for more flexibility in operations is pertinent not only to the auto industry. In the field of public transport, the conflict between standardization and flexibility plays out in the industry's constant struggle to strike a balance between fixed-route service and purely demand-responsive service. For instance, the Hong Kong Airport Express operator provides free shuttle bus service for passengers from Hong Kong and Kowloon train stations to nearby hotels. This is currently served by several fixed-route shuttle services, each serving a dedicated set of hotels (in a fixed sequence) and departing at regular intervals. In low-demand time periods, however, these fixed-route services are not efficient, as the small number of stops could have been served using fewer buses deployed based on the actual destinations of the passengers. A demand-responsive service would be more efficient, but implementation would be challenging. Many public transport operators are nevertheless considering ways to implement flexible transit services in a cost effective manner during periods of low demand[3].

We consider next a concrete challenge faced by the operator of Singapore Changi International Airport, one of the most efficient airports in the world and the seventh busiest international airport. As of March 31, 2013, the airport had served more than 100 airlines operating more than $6,500$ weekly scheduled flights (cf. Table 1), which connect Singapore to more than 250 cities in about 60 countries and territories worldwide. More than $147,000$ passengers pass through its gates daily, bringing with them great challenges to the airport's security-screening operations.

While most airports conduct security screenings in one centralized location, Changi Airport uses a decentralized security screening, in which departing passengers are checked at the entrance to each gate before boarding. Compared with the typical centralized security operation, this decentralized approach helps relieve the bottleneck at the entrance to the transit area (see Figure 1 for a comparison of centralized screening and decentralized screening).

[1] This is practiced by Honda Motor, where all of the vehicles coming into a factory's assembly zones share common designs, such as similar locations and installation techniques for functions like brakes or transmission.

[2] http://www.businessinsider.sg/strategies-that-make-honda-innovative-2014-7#.Vw2qSuJ96Uk

[3] See, for instance, "Operational Experiences with Flexible Transit Services" by David Koffman, published by TRB in 2004

Table 1 Sample flight departure information for Singapore Changi Airport

TEP	FLIGHT	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Dep Time	TO	Plane Type
1	CV785						Y		0005	KUL / GYD	744
3	SQ225	Y	Y	Y	Y	Y	Y	Y	0005	PER	333
3	VA5507	Y	Y	Y	Y	Y	Y	Y	0005	PER	333
3	VS7225	Y	Y	Y	Y	Y	Y	Y	0005	PER	333
3	OZ6782	Y	Y	Y	Y	Y	Y	Y	0010	ICN	333
3	SQ608	Y	Y	Y	Y	Y	Y	Y	0010	ICN	333
1	CA970	Y	Y	Y	Y	Y	Y	Y	0015	PEK	333
2	5J548	Y	Y	Y	Y	Y	Y	Y	0020	CEB	320
1	AF259	Y	Y	Y	Y	Y	Y	Y	0025	CDG	773
1	AZ2691	Y	Y	Y	Y	Y	Y	Y	0025	CDG	773
1	KL836	Y			Y		Y		0030	AMS	772
1	KL836		Y	Y		Y		Y	0030	AMS	773

(a) Decentralized screening in Changi airport (b) Centralized Screening in Israel[4]

Figure 1 Decentralized versus Centralized Screening

The decentralized layout, however, poses a challenge to airport operators. In addition to stationing a regular security screening team at each departure gate, the airport would have to send backup screening teams, known as “roving teams”, to gates facing a surge when passenger arrive, to reduce waiting time. Figure 1a shows the screening operation at one of the boarding gates with a long queue forming, and an adjacent screening line being activated.

Demand for additional roving teams can only arise within roughly one hour before flight departure, and depends on the passenger load for each flight as well as their arrival patterns, which are random. In the case of low passenger load, excess roving teams can be used for other jobs. Therefore, the demand for a roving team at each gate is random and it is either 0 or 1. An interesting but challenging problem faced by the airport is the deployment of roving teams to gates—which are throughout the terminals (see Figure 2 for gate layout)—within a given time frame to meet potential demand so that the fewest total roving teams are required.

[4] The picture is from http://i-hls.com/2015/04/homeland-security-steps-up-screening-of-aviation-employees/

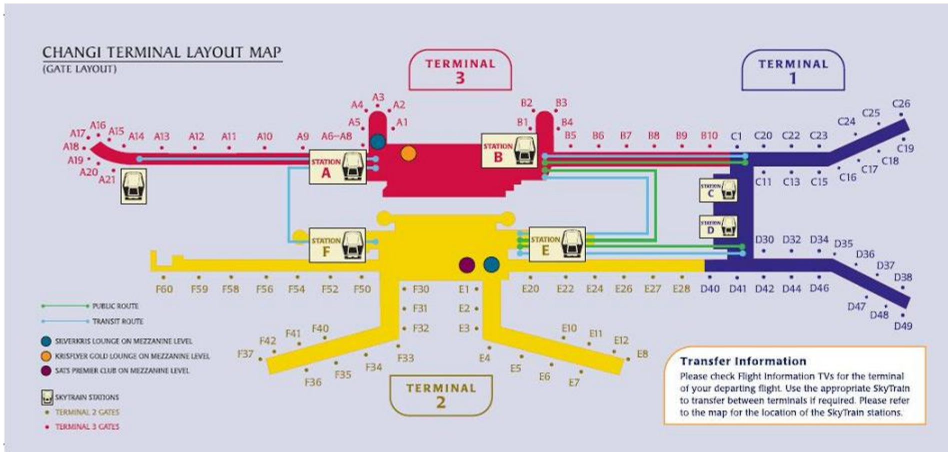


Figure 2 Changi Terminal Layout

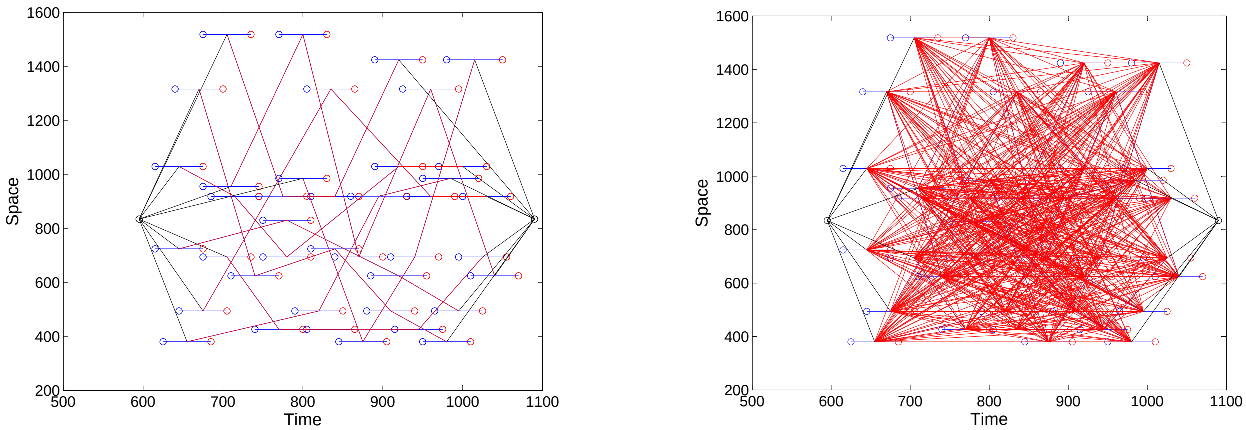
The airport's current deployment approach is to monitor the congestion level at each screening gate through CCTV, and deploy a roving team if a supervisor decides this is necessary. This deployment approach is very flexible. However, this has resulted in a complicated workflow structure in which communication and coordination are challenging; At peak times, even supervisors have to join roving teams.

There is simpler way to address this problem, analogous to a fixed-route service approach, which is to deploy an additional roving team to each gate at the time of departure regardless of the passenger load and their arrival patterns. This ensures stability in the work schedule and simplifies the workflow, but at the expense of maintaining an excessive number of roving teams.

The difference between these two approaches can be illustrated using the following example. Denote the potential demand for a roving team at each boarding gate before flight departure by a node $(x, [t_b, t_e])$, where

- x denotes the location coordinate of the gate, and
- $[t_b, t_e]$ denotes the time interval within which a roving team might be needed.

The simple approach (fixed-route service approach) is based on a standardized structure that reduces the deployment problem to identifying the minimum number of paths covering these demand nodes, with each path representing the schedule of a roving team. A path can cover node $(x(i), [t_b(i), t_e(i)])$, followed by $(x(j), [t_b(j), t_e(j)])$, provided the roving team can traverse from $x(i)$ at time $t_e(i)$ to reach $x(j)$ before $t_b(j)$. The difference between the fixed-route service approach and the current approach (based on a fully flexible structure) for this problem can be understood through the possible workflows of the roving teams throughout the terminal. The fixed-route service approach, as shown in Figure 3, uses a minimum of 9 roving teams, but with a much less complicated workflow compared to the flexible approach, which requires fewer roving teams on average.



(a) Fixed Route Service Approach with 9 roving teams

(b) Fully Flexible Approach with all possible node pairs

Figure 3 Fixed Route Service Approach versus Fully Flexible Approach

While the "complex" workflow structure in the roving team deployment problem does not translate directly into additional operational costs, we note that the problem posed is common in many workforce deployment problems, and the complexity in the deployment structure can be directly related to the additional cost of training workers. For instance, if each node in our network represents a pick-up location and timing in a transportation network, with unknown passenger demand, then our problem reduces to finding the minimum number of vehicles needed to service the (random) demand at each location. This becomes a tanker scheduling problem with random demand (see Ahuja et al. (1995)). Each arc in our network corresponds to a route between two pick-up locations. In more general workforce deployment problems, workers may require different skill sets to serve the demands of different nodes. It is therefore desirable that the system has the sparsest possible workflow structure without sacrificing too much the efficiency of the workforce deployment system.

Bearing in mind our comparison of the two deployment workflow structures—the "standardized" structure and the "fully flexible" structure—for this class of problems, we want to understand if the performance of the standardized structure can be greatly enhanced, meaning the expected number of roving teams can be reduced, by adding a small number of additional arcs to the structure. Specifically, we are looking for a deployment structure that is able to deploy roving teams using a simple workflow comparable to the standardized approach, but with operational efficiency similar to the flexible approach.

This paper provides a methodology to design such a sparse and yet efficient structure. Assuming that demand for roving teams is independent among gates and is 0 or 1 with equal probability, Figure 4 presents the expected number of roving teams needed under different structures, constructed using the methodology developed in this paper. Expected performance is computed using numerical simulation. It is interesting to observe that more than 80% of the arcs can be deleted with negligible impact on system performance.

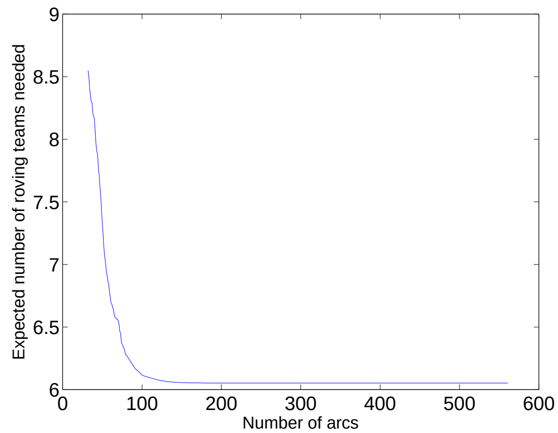
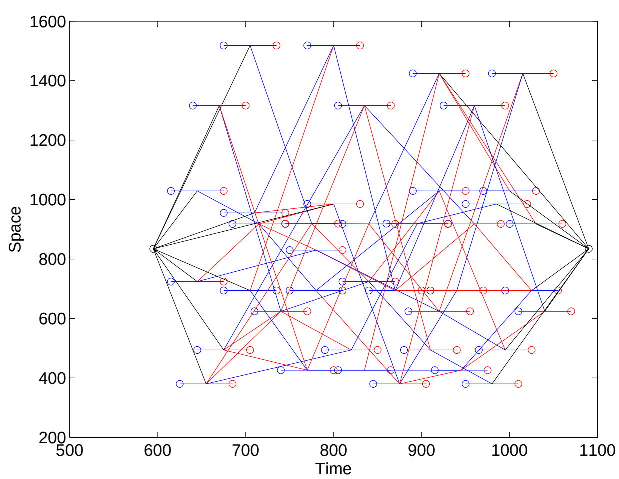
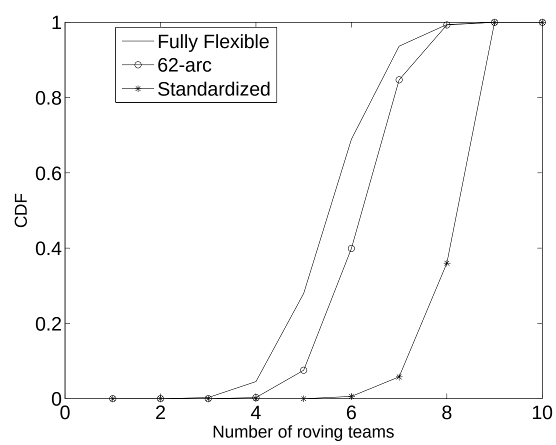


Figure 4 **Expected performance under different structures**

For further illustration, we use the method developed in this paper to construct a 62-arc structure in Figure 5a, with new arcs added to the original fixed route service structure in Figure 3a. The sample path performance in Figure 5b shows that the 62-arc structure performs as well as the fully flexible structure, but with a much smaller number of arcs. A little flexibility does indeed add a lot to roving-team efficiency!



(a) 62-arc structure



(b) Sample path performance under different structures

Figure 5 **Designed sparse but efficient structure**

Jordan and Graves (1995) observed a similar phenomenon in the case of manufacturing process flexibility. They propose that "a small amount of flexibility added in the right way can have virtually all the benefits of total flexibility," and demonstrated this phenomenon in General Motors' manufacturing network. This interesting observation has generated extensive follow-up research on sparse and efficient operations in the field of process flexibility. Jordan and Graves (1995) found the long chain to be a very efficient structure in the case of a symmetric and balanced system with independent and identically distributed (i.i.d.) demand. Following their work, subsequent literature explores good flexible structures in skill chaining and supply chain settings (see Iravani et al. (2005), Deng and Sheng (2013), Simchi-Levi and Wei (2015), etc.) using various ad hoc approaches. Most of these techniques only evaluate and compare two given structures, and could not be used to construct near-optimal process structures.

In general, it is difficult to find a methodology for constructing an efficient structure for the process flexibility problem, not to mention other operational problems in more general settings (e.g., the "roving team" deployment problem faced by Changi Airport). We contribute to the literature in the following ways: First, we contribute to the process flexibility literature by proposing a method to construct an efficient sparse structure for the general problem (including the non-i.i.d demand case). Second, in terms of methodology, we use a completely positive program to reformulate the worst-case model of a network flow problem. Essentially, our model solves the worst-case problem with limited knowledge of demand information (e.g., the first-two moments). The importance of an arc is assessed based on the dual price information in a related conic program. This builds on a recent observation by Wang and Zhang (2015) that a related SDP cone can be used to analyze the worst-case performance of 2-chain in the process flexibility problem, obtaining bounds that are strikingly close to the case in which the demands are normally distributed.

Under general conditions on the moments structure, our dual formulation (a copositive program) satisfies strong duality, so that the associated dual prices provide an estimate of the value of an arc in the system. This is also essential for the numerical technique we use to solve this problem. In the copositive programming literature, strong duality results are known for only several classes of problems (cf. de Klerk and Pasechnik (2002) on the stable set problem).

The rest of the paper is organized as follows: In Section 2, we review the related literature. In Section 3, we present a general distributionally robust model for a stochastic network flow problem. We propose an equivalent completely positive reformulation and further provide a general condition for strong duality between the completely positive program and its dual. Building on the strong duality result, a dual-variable-based heuristic is proposed to design an efficient and sparse network. In Section 4, we apply the framework to tackle two classes of problems: roving team deployment and the classical process flexibility problem. We conclude the paper in Section 5. All proofs are presented in the Appendix.

2. Literature Review

(1) Process Flexibility

The literature on process flexibility is extensive. We now briefly review the papers that are closely related to our studies. Jordan and Graves (1995) produced the classic work on process flexibility, based on General Motors' production process. They put forward the notion that limited flexibility can yield good performance and that the long-chain structure can be nearly as efficient as a fully flexible structure when demand and supply are balanced and symmetric. Following their work, one stream of research analyzes the performance of the chaining structure. Chou et al. (2010) used a random walk method to compute asymptotic expected demand satisfaction in a long-chain structure and compare the long chain's asymptotic performance to that of the full-flexibility structure. Simchi-Levi and Wei (2012) characterized the long chain's performance using the difference between two open chains. Based on this characterization, they proved that the long chain is the optimal structure among 2-flexibility designs. Wang and Zhang (2015) derived a closed-form distribution-free bound on the ratio of the long chain's expected performance relative to that of full flexibility.

Jordan and Graves (1995) used an automobile production example to illustrate how a sparse and efficient structure (an additional 6-arc structure) can be constructed for the general process flexibility problem. Chou et al. (2011) showed that when demand is bounded, there exists a set of graph expanders that can achieve near-optimal performance compared to the full-flexibility structure. They also proposed a heuristic to find such structures. Desir et al. (2015) found that for some instances of demand distribution, a disconnected network performs strictly better than the long chain. They further proved that the long chain is indeed optimal among connected networks.

There are various ways to "index" the performance of a process structure. Jordan and Graves (1995) developed a probabilistic index as a surrogate to measure the performance of any process structure. The probabilistic index is defined as the largest probability among all the demand node subsets whose probability of unsatisfied demand would exceed that in the full-flexibility counterpart. Iravani et al. (2005) introduced a "structural flexibility matrix" to evaluate a systems's process capability. They used the means of matrix entries and the largest eigenvalue as flexibility indices. Chou et al. (2008) proposed to use the second smallest eigenvalue of the Laplacian matrix as the "expansion index" to design a flexible structure. Simchi-Levi and Wei (2015) extended the study from a worst-case perspective and introduced "plant cover index" as a measure of worst-case performance. The plant cover index is able to compare two flexibility designs' worst-case performances based only on the design structures, regardless of the uncertainty set. Note that these indices work when a structure is given, but they cannot be used to guide the design of good process structure.

(2) The Workforce Deployment Problem

The workforce composition and deployment problem is complicated, as it involves workers with different skill sets and sometimes combines full-time workers and part-time workers. According to the Bureau of Labor Statistics (2013), 70% of organizations in U.S. employ part-time workers, which demonstrates the prevalence of flexible labor resources in today's economy. Kesavan et al. (2014) studied flexible labor resources' impact on a firm's financial performance. They used flexible labor resources to adjust capacity accordingly to meet demand, and found through empirical methods that a mixture of full-time and part-time workers shows an inverted U-shaped relationship with sales performance and profit, and a U-shaped relationship with expenses. However, the paper did not address how flexible labor forces can be deployed in practice.

A key difficulty in making deployment decisions is uncertain demand. Nobert and Roy (1998) studied the schedule of freight-handling personnel at air cargo terminals. They used demand forecasts to determine the amount of freight ready to be handled each day and further introduced a new approach called "demand leveling" to identify the true demand for manpower. Bard et al. (2007) approximated daily demand using three-point distribution based on historical data from United States Postal Service (USPS) mail processing and distribution centers (P&DCs), and developed a two-stage stochastic integer program with recourse to analyze the effect of deployment on labor cost. They considered both full- and part-time workers, and obtained the number of each type of worker needed. Zhu and Sherali (2009) considered both long-term demand fluctuation and short-term demand uncertainty. They used expected demand profile over the horizon to hedge against long-term demand fluctuation. For short-term demand uncertainty, they proposed a two-stage stochastic program and applied a Bender's decomposition-based algorithm to solve the two-stage model. For a comprehensive survey on workforce deployment, we refer the reader to Bergh et al. (2013).

Qin et al. (2015) provided a thorough review of the workforce flexibility literature, grouped five workforce-flexibility methods—flexible working hours, floaters, cross-training, teamwork, and temporary labor. Labor flexibility allows the system to dynamically reallocate resources from one stage of production to another in response to shifting bottlenecks. For instance, Daniels et al. (2004) studied the value of partial workforce flexibility in a flow shop scheduling environment and reported that "a large fraction of the benefit of complete flexibility can be obtained with a relatively modest amount of partial flexibility (p.8)."

(3) Conic Programming Approach A completely positive cone is defined as

$$\begin{aligned}\mathcal{CP}_n &:= \{A \in S_n | \exists V \in \mathcal{R}_+^{n\times m}, \text{ such that } A = VV^T\}\\ &:= \{A \in S_n | \exists \mathbf{v_1}, \mathbf{v_2}, ..., \mathbf{v_k} \in \mathcal{R}_+^n, \text{ such that } A = \sum_{i=1}^k \mathbf{v_i}\mathbf{v_i^T}\}\end{aligned}$$

where S_n is the set of $n \times n$ symmetric matrices.

A copositive cone is the dual cone of a completely positive cone. It is defined as

$$\mathcal{CO}_n := \{A \in S_n | \forall \mathbf{v} \in \mathcal{R}^n_+, \mathbf{v}^\mathsf{T}\mathbf{A}\mathbf{v} \geq 0\}$$

A completely positive program is defined as a linear program over a completely positive cone. Its dual problem—copositive program—is defined as a linear program over the dual cone, or a copositive cone. Completely positive and copositive programs have recently been used to model $\mathcal{NP}$-hard optimization problems. It is well known that a maximal stable set problem cannot be approximated within a factor $|V|^{\frac{1}{2}-\epsilon}$ for any $\epsilon > 0$ unless $\mathcal{P} = \mathcal{NP}$. Lovasz and Schrijver (1991) linked the stability number with the so-called theta number, defined as the optimal value of a related semidefinite program. de Klerk and Pasechnik (2002) extended this work and showed that by changing the semidefinite cone to a copositive cone, the resulting copositive program is exact for the stable set problem. They also showed how to approximate the copositive cone using a hierarchy of linear or semidefinite programs to compute the stability number. Interestingly, it will be shown later in the paper that the dual of the classic maximum flow model for solving process flexibility problem can be regarded as a "random-weight stable set" problem. In this paper, we equivalently reformulate this type of random-weight stable set problem as a completely positive program and show that the strong duality holds under the proposed reformulation.

Burer (2009) showed the equivalence between completely positive problems and nonconvex quadratic problems with a mixture of binary and continuous variables, which is a well-known $\mathcal{NP}$-hard problem. Kong et al. (2013) applied this approach to health-care appointment scheduling problems, demonstrating the potential of copositive programming to solve difficult operations management problems. The closest related literature to our work is Natarajan et al. (2011), who presented an equivalent completely positive reformulation to mixed 0-1 linear programming problems with random objectives. We extend their model to incorporate quadratic constraints in the formulation, and show how these constraints are related to sensitivity analysis of underlying operational problems.

It is worth mentioning that reformulating an $\mathcal{NP}$-hard problem to a completely/copositive program does not resolve the underlying difficulty in computation, but it helps to shift the combinatorial complexity to the facial structure of general completely/copositive cones. For more information on completely positive cones and copositive cones, we refer readers to Berman and Shaked-Monderer (2003). Although in general solving a completely positive program or a copositive program is $\mathcal{N}P$-hard, Natarajan and Teo (2016) prove that in some special cases the competely positive program can be equivalently solved exactly by an SDP. In general cases, several approximation methods—so-called hierarchies, which involve a sequence of tractable cones—have been studied. Gaddum (1958) proposed an approach for checking whether a cone is copositive (which he termed "conditionally semi-definite") by solving linear programs. In the meantime, SDP-based hierarchies have become popular in the literature. One method is to approximate a completely positive cone as a "doubly nonnegative cone," which is defined as a positive semi-definite cone with nonnegative entries. Specifically, $\{A|A \succcurlyeq 0, A \geq 0\}$ provides a outer approximation for completely positive cones, and $\{A|A = A_1 + A_2, A_1 \succcurlyeq 0, A_2 \geq 0\}$ gives an inner approximation for copositive cones (de Klerk and Pasechnik (2002), Parrilo (2000)). In this paper, we use doubly nonnegative cones to approximate completely positive or copositive cones.

The use of algorithms to solve a doubly nonnegative matrix has received a lot of attention in recent years. Sun et al. (2015) employ a majorized semi-smooth Newton-CG augmented Lagrangian method, coupled with a convergent 3-block alternating direction method of multipliers, to obtain a solution with moderate accuracy. They also develop a software, SDPNAL+, that can be used to solve moderate- to large-sized doubly nonnegative problems efficiently, as long as strong duality holds. We refer the reader to Yang et al. (2014) for details on the software, as well as some impressive computational results. In our paper, we use this software to solve the doubly nonnegative programs.

3. A Distributionally Robust Model of Stochastic Network Flow and Sparse Design

A stochastic network flow model has been widely used in modeling operations management problems in an uncertain environment. In this section, we develop a general distributionally robust model of a stochastic network flow problem and establish the model's key properties for the sparse design problem.

Let $\mathcal{G}(\mathcal{V}_0, \mathcal{A}_0)$ denote a network with node set $\mathcal{V}_0$ and arc set $\mathcal{A}_0$. Denote the number of nodes in the network as n, i.e., $|\mathcal{V}_0| = n$. By adding two virtual nodes—one source node s, one sink node t—and linking them to all nodes in $\mathcal{V}_0$, we construct a new network, $\mathcal{G}(\mathcal{V}, \mathcal{A})$. Given this network, consider a min-cost-flow problem with random demand $\tilde{\mathbf{d}}$ at each node in $\mathcal{V}_0$:

$$
\begin{aligned}
Z(\tilde{\mathbf{d}}) = \min_{x_{ij}} \quad & \sum_{(i,j)\in\mathcal{G}(\mathcal{V},\mathcal{A})} c_{ij}x_{ij} \\
s.t. \quad & \sum_{i\in\mathcal{V}_0\bigcup\{s\},(i,j)\in\mathcal{A}} x_{ij} \geq \tilde{d}_j, \quad j \in \mathcal{V}_0 \\
& \sum_{i\in\mathcal{V}_0\bigcup\{s\},(i,j)\in\mathcal{A}} x_{ij} - \sum_{i\in\mathcal{V}_0\bigcup\{t\}(j,i)\in\mathcal{A}} x_{ji} = 0, \quad j \in \mathcal{V}_0 \\
& x_{ij} \geq 0,
\end{aligned} \tag{1}
$$

where c_{ij} is the cost of unit flow for arc (i,j). Note that the results developed hold also for the following equivalent model:

$$
\begin{aligned}
Z(\tilde{\mathbf{d}}) = \max_{x_{ij}} \quad & \sum_{(i,j)\in\mathcal{G}(\mathcal{V},\mathcal{A})} c_{ij}x_{ij} \\
s.t. \quad & \sum_{i\in\mathcal{V}_0\bigcup\{s\},(i,j)\in\mathcal{A}} x_{ij} \leq \tilde{d}_j, \quad j \in \mathcal{V}_0 \\
& \sum_{i\in\mathcal{V}_0\bigcup\{s\},(i,j)\in\mathcal{A}} x_{ij} - \sum_{i\in\mathcal{V}_0\bigcup\{t\}(j,i)\in\mathcal{A}} x_{ji} = 0, \quad j \in \mathcal{V}_0 \\
& x_{ij} \geq 0,
\end{aligned} \tag{2}
$$

For ease of exposition, we focus the development of the theory for Model (1).

Due to the strong duality of linear programming, by introducing dual variables $\mathbf{y}, \mathbf{z}$ for the two sets of constraints in Problem (1), we can get the corresponding dual formulation as follows:

$$Z(\tilde{\mathbf{d}}) = \max_{\mathbf{y},\mathbf{z}\in\mathcal{X}} \sum_{j\in\mathcal{V}_0} \tilde{d}_j y_j \tag{3}$$

Where

$$\mathcal{X} = \left\{ \begin{array}{rlc} y_j + z_j \le & c_{sj}, & j \in \mathcal{V}_0, (s,j) \in \mathcal{A} \\ y_j + z_j - z_i \le & c_{ij}, & (i,j) \in \mathcal{A}_0 \\ -z_i \le & c_{it}, & i \in \mathcal{V}_0, (i,t) \in \mathcal{A} \\ \mathbf{y} \ge & \mathbf{0} & \end{array} \right\}$$

We adopt the concept of a distributionally robust optimization and study the worst-case expected value of $Z(\tilde{\mathbf{d}})$ assuming we only have the first-two moments information of $\tilde{\mathbf{d}}$. Specifically, we assume that the distribution of $\tilde{\mathbf{d}}$ lies in a set of multivariate distributions supported on nonnegative orthant of dimension n, $\mathbb{R}^n_+$, with finite first moment $\boldsymbol{\mu}_d$ and finite second moment Σ_d. Then the worst-case expected value of $Z(\tilde{\mathbf{d}})$ is obtained by solving

$$Z_P = \sup_{\tilde{\mathbf{d}}\sim(\boldsymbol{\mu}_d,\Sigma_d)} \boldsymbol{E}[Z(\tilde{\mathbf{d}})]. \tag{4}$$

We will show in the next section that Problem (4) can be equivalently reformulated as a conic program. The first step leading to the equivalent reformulation is to rewrite the inner LP problem in a standard form. Notice that to get the constraint "$y_j + z_j - z_i \le c_{ij}$" into standard form, we need to introduce $|\mathcal{A}_0|$ number of slack variables. The corresponding decision variable matrix in the reformulated conic program, based on the approach by Burer (2009), would be of dimension $\mathcal{O}(|\mathcal{A}_0|^2)$. This hinders computational efficiency, especially when the network is large. In this paper, we will exploit special network and cost structures to reduce the dimension to $\mathcal{O}(|\mathcal{A}_0|)$ by appropriate reformulation.

Specifically, in the rest of the paper, we will consider the problem of a special cost structure such that:

$$\begin{cases} c_{it} = 0 & \forall i \in \mathcal{V}_0 \\ c_{ij} \in \{0,1\} & i \in \mathcal{V}_0 \cup \{s\}, j \in \mathcal{V}_0. \end{cases} \tag{5}$$

Since Problem (3) is totally unimodular, the optimal solutions are binary. Therefore, $1 - z_i \in \{0,1\}$. Note that $y_j + z_j \in \{0,1\}$ follows from $y_j + z_j \le c_{sj} \le 1$. We can re-write $y_j + z_j - z_i \le c_{ij}$ in the following way:

$$(y_j + z_j) + (1 - z_i) + (1 - c_{ij}) \le 2, \quad (i,j) \in \mathcal{A}_0.$$

This is equivalent to $(1-c_{ij})(y_j+z_j)(1-z_i) = 0, (i,j) \in \mathcal{A}_0$ since each term is shown to be binary in the optimal solution. Therefore, $\mathcal{X}$ can be equivalently reformulated as

$$\mathcal{X} = \left\{ \begin{array}{rcc} y_j + z_j + s_j = & c_{sj}, & j \in \mathcal{V}_0 \\ (1-c_{ij})(y_j+z_j)(1-z_i) = & 0, & (i,j) \in \mathcal{A}_0 \\ \mathbf{y}, \mathbf{z}, \mathbf{s} \in & \{0,1\}^n & \end{array} \right\}. \tag{6}$$

This reformulation reduces the model to system of equalities, using $\mathcal{O}(|\mathcal{A}_0|)$ number of variables, at the expense of introducing quadratic constraints. There is another important reason to reformulate the linear constraints, $y_j + z_j - z_i \leq c_{ij},\ (i,j) \in \mathcal{A}_0$, into a quadratic form - the dual to these quadratic constraints can be used to design natural arc-selection heuristic for our problems. We will address this explicitly when we introduce the dual-variable-based heuristic.

3.1. Completely Positive Reformulation

We write the quadratic program in a general form as

$$\begin{aligned} Z(\tilde{\mathbf{d}}) = \max_{\mathbf{x}} \quad & \tilde{\mathbf{d}}^\intercal \mathbf{x} \\ s.t. \quad & \mathbf{a}_i^\intercal \mathbf{x} = b_i, \forall i \\ & (\mathbf{h}_i^\intercal \mathbf{x} + f_i)(\hat{\mathbf{h}}_j^\intercal \mathbf{x} + \hat{f}_j) = 0, \forall (i,j) \in \mathcal{H} \\ & x_i \in \{0,1\}, \forall i \in \mathcal{B}, \end{aligned} \tag{7}$$

where $\mathcal{B}$ is the set of indices for the binary variables. $\mathcal{H} = \mathcal{A}_0 \cap \{(i,j) \mid c_{ij} = 0\}$ in our network flow model. We use N to represent the dimension of decision vector, i.e., $\mathbf{x} \in \mathbb{R}^{\mathbf{N}}$ and assume the number of linear constraints in (7) is M.

The key assumptions under which the problem is analyzed are as follows:

A1. $\mathbf{x} \in \Phi := \left\{\mathbf{x} \geq 0 \mid \mathbf{a}_i^\intercal \mathbf{x} = b_i,\ \forall i\right\} \Rightarrow \mathbf{h}_i^\intercal \mathbf{x} + f_i \geq 0, \forall i,\ \hat{\mathbf{h}}_j^\intercal \mathbf{x} + \hat{f}_j \geq 0, \forall j$ and $\mathbf{x} \leq \mathbf{1}$.

A2. The random coefficient $\tilde{\mathbf{d}}$ is defined in a nonnegative support $\mathbb{R}_+^N$, with finite first-two moments $\boldsymbol{\mu}_d$ and Σ_d.

A3. The feasible region is nonempty and bounded.

Assumption A2 and A3 are standard in the literature (c.f. Natarajan et al. (2011), Kong et al. (2013)). Assumption A1 ensures that Φ is sufficient to induce additional non-negativity constraints that will facilitate the reformulation using copositive cones. For instance, in (6), since $c_{sj} \in \{0,1\}$, $y_j + z_j + s_j = c_{sj}, \forall j \in \mathcal{V}_0$, $\mathbf{y},\mathbf{z},\mathbf{s} \geq \mathbf{0}$ implies that $y_j + z_j \geq 0$ and $1 - z_i \geq 0$, and $y_j, z_j, s_j \leq 1$.

For ease of exposition, we first define some notations used in the next part of this section.

- $\mathbf{e}_i \in \mathbb{R}^N$ denotes a unit vector with ith element equal to 1.
- $\mathbf{1}_N \in \mathbb{R}^N$ denotes a vector with all elements equal to 1.
- $\mathbf{0}_N \in \mathbb{R}^N$ denote a vector with all elements equal to 0.
- $J \in \mathbb{R}^{N \times N}$ denotes a matrix with all elements equal to 1.
- $I \in \mathbb{R}^{N \times N}$ denotes the identity matrix.
- $\bullet$ represents the inner product of matrices: $A \bullet B$ denotes the trace of matrix $A^\intercal B$.

Define $\mathbf{x}(\mathbf{d})$ to be the optimal solution to (7) under a specific $\mathbf{d}$. Since $\tilde{\mathbf{d}}$ is a random variable with finite first-two moments $(\mu_{\mathbf{d}}, \boldsymbol{\Sigma}_{\mathbf{d}})$, $\mathbf{x}(\tilde{\mathbf{d}})$ is also a random variable with well-defined first-two moments. Then we define

$$\begin{aligned} \mathbf{p} &:= \mathbf{E}[\mathbf{x}(\tilde{\mathbf{d}})] \in \mathbb{R}_+^N \\ Y &:= \mathbf{E}[\mathbf{x}(\tilde{\mathbf{d}})\tilde{\mathbf{d}}^\intercal] \in \mathbb{R}_+^{N \times N} \\ X &:= \mathbf{E}[\mathbf{x}(\tilde{\mathbf{d}})\mathbf{x}(\tilde{\mathbf{d}})^\intercal] \in \mathbb{R}_+^{N \times N} \end{aligned} \tag{8}$$

where all the expectations are taken with respect to $\tilde{\mathbf{d}}$. Then we have

$$\mathbf{E}\left[\begin{pmatrix}1\\ \tilde{\mathbf{d}}\\ \mathbf{x}(\tilde{\mathbf{d}})\end{pmatrix}\begin{pmatrix}1\\ \tilde{\mathbf{d}}\\ \mathbf{x}(\tilde{\mathbf{d}})\end{pmatrix}^{\mathsf{T}}\right] = \begin{pmatrix}1 & \boldsymbol{\mu}_d^{\mathsf{T}} & \mathbf{p}^{\mathsf{T}}\\ \boldsymbol{\mu}_d & \Sigma_d & Y^{\mathsf{T}}\\ \mathbf{p} & Y & X\end{pmatrix}$$

It is clear that $\begin{pmatrix}1 & \boldsymbol{\mu}_d^{\mathsf{T}} & \mathbf{p}^{\mathsf{T}}\\ \boldsymbol{\mu}_d & \Sigma_d & Y^{\mathsf{T}}\\ \mathbf{p} & Y & X\end{pmatrix}$ is a completely positive matrix by definition.

Following the same technique in Natarajan et al. (2011), the objective, $\mathbf{E}[Z(\tilde{\mathbf{d}})] = \mathbf{E}[\tilde{\mathbf{d}}^{\mathsf{T}}\mathbf{x}(\tilde{\mathbf{d}})]$, can be rewritten as $I \bullet Y$; the linear constraints, $\mathbf{a}_i^{\mathsf{T}}\mathbf{x}(\tilde{\mathbf{d}}) = b_i, \forall i$, implies two set of constraints in the expectation:

$$\begin{aligned}\mathbf{a}_i^{\mathsf{T}}\mathbf{p} &= b_i, \quad \forall i\\ \mathbf{a}_i^{\mathsf{T}}X\mathbf{a}_i &= b_i^2, \ \forall i\end{aligned}$$

The first constraint is obtained by taking the expectation over $\tilde{\mathbf{d}}$, i.e.,

$$\mathbf{E}[\mathbf{a}_i^{\mathsf{T}}\mathbf{x}(\tilde{\mathbf{d}})] = b_i \Longrightarrow \mathbf{a}_i^{\mathsf{T}}\mathbf{p} = b_i, \forall i$$

We use a "lifting" technique to get the second constraint in the following way: Since we have $(\mathbf{a}_i^{\mathsf{T}}\mathbf{x}(\tilde{\mathbf{d}}))(\mathbf{x}(\tilde{\mathbf{d}})^{\mathsf{T}}\mathbf{a}_i) = b_i^2$, by taking the expectation with respect to $\tilde{\mathbf{d}}$, we have $\mathbf{a}_i^{\mathsf{T}}X\mathbf{a}_i = b_i^2$ (c.f. Natarajan et al. (2011)). The binary constraint $\mathbf{x}(\tilde{\mathbf{d}}) \in \{0,1\}^{|\mathcal{B}|}$ implies that $x_i(\tilde{\mathbf{d}}) = x_i(\tilde{\mathbf{d}})^2, \forall i \in \mathcal{B}$. We reformulate this set of quadratic constraints, together with $(\mathbf{h}_i^{\mathsf{T}}\mathbf{x}(\tilde{\mathbf{d}}) + f_i)(\hat{\mathbf{h}}_j^{\mathsf{T}}\mathbf{x}(\tilde{\mathbf{d}}) + \hat{f}_j) = 0, \forall (i,j) \in \mathcal{H}$, by taking the expectation with respect to $\tilde{\mathbf{d}}$. In this way, we have

$$\begin{aligned}X_{ii} &= p_i, && \forall i \in \mathcal{B}\\ \mathbf{h}_i^{\mathsf{T}}X\hat{\mathbf{h}}_j + (f_i\hat{\mathbf{h}}_j^{\mathsf{T}} + \hat{f}_j\mathbf{h}_i^{\mathsf{T}})\mathbf{p} + f_i\hat{f}_j &= 0, && \forall (i,j) \in \mathcal{H}\end{aligned}$$

Consider the following completely positive program with $\mathbf{p}$, X, and Y as decision variables, which are defined in (8) :

$$\begin{array}{llcll} Z_C = \max & I \bullet Y & & & \\ \quad s.t. & \mathbf{a}_i^{\mathsf{T}}\mathbf{p} & = & b_i, & \forall i = 1,\ldots,M\\ & \mathbf{a}_i^{\mathsf{T}}X\mathbf{a}_i & = & b_i^2, & \forall i = 1,\ldots,M\\ & X_{ii} & = & p_i, & \forall i \in \mathcal{B}\\ & \mathbf{h}_i^{\mathsf{T}}X\hat{\mathbf{h}}_j + (f_i\hat{\mathbf{h}}_j^{\mathsf{T}} + \hat{f}_j\mathbf{h}_i^{\mathsf{T}})\mathbf{p} + f_i\hat{f}_j & = & 0, & \forall (i,j) \in \mathcal{H}\\ & CP = \begin{pmatrix}1 & \boldsymbol{\mu}^{\mathsf{T}} & \mathbf{p}^{\mathsf{T}}\\ \boldsymbol{\mu} & \Sigma & Y^{\mathsf{T}}\\ \mathbf{p} & Y & X\end{pmatrix} & \succcurlyeq_{cp} & 0, \ \boldsymbol{\mu} = \mu_{\mathbf{d}}, \ \boldsymbol{\Sigma} = \boldsymbol{\Sigma}_{\mathbf{d}}. & \end{array} \tag{9}$$

From the construction of this completely positive program, it is clear that (9) is a relaxation to problem (4), hence $Z_C \geq Z_P$. In the following proposition, we will show that these two formulations are, in fact, equivalent.

PROPOSITION 1. *The completely positive program Z_C is equivalent to the worst-case model Z_P, i.e., $Z_C = Z_P$.*

To show the equivalence, we adopt a similar proof technique used in the literature (cf. Natarajan et al. (2011)). We construct a sequence of distributions that satisfies the moment conditions in the limit and show that the limit of the set of feasible solutions under such a distribution sequence achieves the upper bound provided by Z_C. See Appendix A for the detailed proof.

Remark: The framework can incorporate the additional condition that the random cost coefficients are binary, i.e., from the Bernoulli family of distributions. According to Burer (2009), the valid moments from this family can be characterized using a set of completely positive constraints. i.e., the moments $(\mu, \mathbf{\Sigma})$ are feasible moments for a Bernoulli family of distributions if and only if the following set is not empty :

$$\Omega(\boldsymbol{\mu}, \Sigma) = \left\{ (\mathbf{w}, W) \;\middle|\; \begin{array}{ll} \exists \mathbf{s}, S, Y \text{ such that} & \\ \mathbf{w} = \boldsymbol{\mu}; & W = \Sigma; \\ w_i = W_{ii}, \forall i = 1, ..., N; & s_i = S_{ii}, \forall i = 1, ..., N; \\ w_i + s_i = 1, \forall i = 1, ..., N; & W_{ii} + S_{ii} + 2Y_{ii} = 1, \forall i = 1, ..., N; \\ \begin{pmatrix} 1 & \mathbf{w}^\intercal & \mathbf{s}^\intercal \\ \mathbf{w} & W & Y \\ \mathbf{s} & Y^\intercal & S \end{pmatrix} \succcurlyeq_{cp} 0 & \end{array} \right\} \tag{10}$$

This provides the necessary and sufficient conditions to characterize the family of Bernoulli distributions with given first-two moments.

3.2. Strong Duality

In this paper, the proposed heuristic to design an efficient sparse structure is based on dual variables in the completely positive program (9). Hence, before moving on to the heuristic, we first explore the conditions under which strong duality of Problem (9) holds. We first construct the dual formulation of (9), which is a copositive program.

Denote the dual variables corresponding to each set of linear constraints in (9) as $\boldsymbol{\beta}^{(1)}, \boldsymbol{\beta}^{(2)} \in \mathbb{R}^M$, $\boldsymbol{\beta}^{(3)} \in \mathbb{R}^{|\mathcal{B}|}$, and $\Gamma \in \mathbb{R}^{n \times n}$ (where n denote number of vertices), following the sequence of their presentation in (9). Notice that in the completely positive matrix CP,

$$CP_{1,1} = 1;$$

$$CP_{1,(j+1)} = CP_{(j+1),1} = \mu_{dj}, \forall j = 1, ..., N;$$

and

$$CP_{(i+1),(j+1)} = \Sigma_{dij}, \forall i, j = 1, ..., N.$$

We denote the dual variables corresponding to these moment constraints as $\alpha_0 \in \mathbb{R}$, $\boldsymbol{\beta_0} \in \mathbb{R}^N$, and $\Gamma_0 \in \mathbb{R}^{N \times N}$. Define

$$\begin{aligned} \mathbf{w} = & \quad \sum_{i=1}^{M} \beta_i^{(1)} \mathbf{a}_i - \sum_{i \in \mathcal{B}} \beta_i^{(3)} \mathbf{e}_i + \sum_{(i,j) \in \mathcal{H}} \Gamma_{ij} (f_i \hat{\mathbf{h}}_j + \hat{f}_j \mathbf{h}_i) \\ W = & \sum_{i=1}^{M} \beta_i^{(2)} \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in \mathcal{B}} \beta_i^{(3)} \mathbf{e}_i \mathbf{e}_i^\top + \sum_{(i,j) \in \mathcal{H}} \Gamma_{ij} (\tfrac{1}{2} \hat{\mathbf{h}}_j \mathbf{h}_i^\top + \tfrac{1}{2} \mathbf{h}_i \hat{\mathbf{h}}_j^\top) \end{aligned}$$

Then the dual problem of (9) can be written as

$$\begin{aligned} Z_{CD} = & \min \ \alpha_0 + \boldsymbol{\mu}^\top \boldsymbol{\beta}_0 + \Sigma \bullet \Gamma_0 + \mathbf{b}^\top \boldsymbol{\beta}^{(1)} + \mathbf{b}^\top diag(\boldsymbol{\beta}^{(2)}) \mathbf{b} - \sum_{i,j} \Gamma_{ij} \hat{f}_j f_i \\ & s.t. \qquad \begin{pmatrix} \alpha_0 & \frac{1}{2}\boldsymbol{\beta_0}^\top & \frac{1}{2}\mathbf{w}^\top \\ \frac{1}{2}\boldsymbol{\beta_0} & \Gamma_0 & O \\ \frac{1}{2}\mathbf{w} & O & W \end{pmatrix} - \begin{pmatrix} 0 & \mathbf{0_N}^\top & \mathbf{0_N}^\top \\ \mathbf{0_N} & O & \frac{1}{2}I \\ \mathbf{0_N} & \frac{1}{2}I & O \end{pmatrix} \succcurlyeq_{co} 0 \end{aligned} \tag{11}$$

where *diag* is an operator that maps a vector to a diagonal matrix with vector elements as the diagonal elements of the matrix; and O stands for 0 matrices of proper dimension.

We establish strong duality using Slater constraint qualification, i.e., as long as there is a feasible solution $(\mathbf{p}, Y, X)$ to Problem (9) that lies in the interior of the completely positive cone defined in (9), there is no duality gap between Z_C and Z_{CD}. Notice that Dickinson (2010) has characterized the interior of a $N \times N$-dimensional completely positive cone, denoted as $\mathcal{C}_N^*$:

$$int(\mathcal{C}_N^*) = \left\{ \sum_{i=1}^{m} \mathbf{a}_i \mathbf{a}_i^\top : \begin{array}{c} \mathbf{a}_i \in \mathbb{R}_+^N, \forall i = 1, ..., m \\ \text{span}\{\mathbf{a}_1, \mathbf{a}_2, ...\mathbf{a}_m\} = \mathbb{R}^N, \\ \exists \mathbf{a} \in \{\mathbf{a}_1, ...\mathbf{a}_m\} \quad \text{such that} \quad \mathbf{a} > \mathbf{0} \end{array} \right\} \tag{12}$$

Based on this characterization, we provide conditions for the strong duality between a completely positive program (9) and its dual formulation (11) in the following theorem.

THEOREM 1. *Suppose the following conditions hold:*

(i) The moment matrix $\begin{pmatrix} 1 & \boldsymbol{\mu}_d^\top \\ \boldsymbol{\mu}_d & \Sigma_d \end{pmatrix}$ *lies in the interior of a* $(1+N) \times (1+N)$*-dimensional completely positive cone* $\mathcal{C}_{1+N}^*$*;*

(ii) There exists a set of feasible solutions $\mathbf{x}^{(i)}, i = 1, \ldots, m$ *to Problem (7), such that span* $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\} = \mathbb{R}^N$ *and at least one of them is strictly positive, i.e.,* $\exists \mathbf{x}^{(l)} \in \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ *such that* $\mathbf{x}^{(l)} > 0$.

Then strong duality holds between the completely positive program (9) and its dual formulation (11), i.e., $Z_C = Z_{CD}$.

Condition (ii) requires Problem (7) to admit a strictly positive interior solution. Unfortunately, this condition often fails to hold. For instance, to turn inequality constraints into equality constraints in our approach, we need to add additional slack variables. In the optimal solutions, some of the inequalities are binding so that the corresponding slack variables have to be 0. To resolve the issue brought by the slack variables, we divide the solution $\mathbf{x}$ into two parts.

One is composed of decision variables θ, and the other includes all the slack variables $\mathbf{s}$, i.e., $\mathbf{x} = \begin{pmatrix} \boldsymbol{\theta} \\ \mathbf{s} \end{pmatrix}, \boldsymbol{\theta} \in \mathbb{R}^{n_1}, \mathbf{s} \in \mathbb{R}^{N-n_1}$. We modify Condition(ii) in Theorem 1 to

(ii') There exists a set of feasible solutions $\begin{pmatrix} \boldsymbol{\theta}^{(i)} \\ \mathbf{s}^{(i)} \end{pmatrix}$ to (7) such that span$\{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(m)}\} = \mathbb{R}^{n_1}$ and at least one of them is strictly positive, i.e., $\exists \boldsymbol{\theta}^{(l)} \in \{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(m)}\}$ such that $\boldsymbol{\theta}^{(1)} > 0$.

Theorem 2. *Under Conditions (i) and (ii'), there is no duality gap between the completely positive program (9) and its dual formulation (11), i.e.,* $Z_C = Z_{CD}$.

The proof of Theorem 2 is based on the generalized Slater constraint qualification, i.e., as long as there is a feasible solution $(\mathbf{p}, Y, X)$ to Problem (9) that lies in the relative interior of the completely positive cone defined in (9), there is no duality gap between Z_C and Z_{CD}. Compared to Theorem 1, the main challenge in proving the strong duality stems from the construction of a relative interior point. We tackle this issue by starting with a construction without slack variables—i.e., construct a strict interior solution in a smaller completely positive cone based on the set of $\boldsymbol{\theta}^{(i)}$ satisfying Condition (ii'). Then we map the interior point into a relative interior point of a completely positive cone in higher dimension, which is the original cone given in (9). See Appendix A for details of the formal proof.

3.3. Dual-Variable-Based Heuristic

We propose a dual-variable-based heuristic to design sparse network structure. We start with a full graph—namely, the network with all the possible arcs linking the nodes in $\mathcal{V}_0$, as well as the ones linking with source node s and sink node t. We use $\mathcal{G}^F$ to denote this network. The main idea of the heuristic is to incrementally delete the arc with the smallest absolute value in the optimal dual solution. We motivate this heuristic based on the observation that a bound on the change in Z_{CD} is related to the magnitude of the dual variable in the conic programming model. Formally,

Proposition 2. *Under the cost structure specified in (5), the increase in* Z_{CD} *after deleting arc* $(a,b) \in \mathcal{G}$ *is bounded above by* $(\frac{1}{4} + c_{sb})|\Gamma^*_{ab}|$.

Note that c_{sb} is the cost of flow on arc (s,b), and Γ^*_{ab} is the (a,b)-th entry of the optimal dual matrix Γ^*. We prove this proposition via a constructive approach. We first construct a feasible solution to (11) in the network after deleting arc (a,b). Using this feasible solution, we show that the objective value Z_{CD} increased by not more than $(\frac{1}{4} + c_{sb})|\Gamma^*_{ab}|$.

From Proposition 2, we can see clearly why we need quadratic constraints in the reformulation of (3). If we use the original linear constraints "$y_j + z_j - z_i \le c_{ij}$," or its standard form after adding slack variable s_{ij}, the corresponding equivalent completely positive program will have two sets of linear constraints, one from taking the direct expectation, i.e., "$E[y_j + z_j - z_i + s_{ij}] = E[c_{ij}]$," and the other from taking the expectation of the lifted linear constraint "$E[(y_j + z_j - z_i + s_{ij})^2] = E[c_{ij}^2]$." In that case, it is not clear which dual variable will be more appropriate to indicate the vaue of an arc in the network flow problem. However, by reformulating the original linear constraints into a quadratic form, there is only one unique constraint in (9) that corresponds to each arc. Therefore, the dual variable of the constraint in (9) can provide a unique arc-selection criterion in our heuristic.

It is worth mentioning that, despite the nice property of the dual variable in the copositive program, solving a copositive program is, in general, an $\mathcal{NP}$-hard problem (cf. Murty and Kabadi (1987)). In this paper, we use instead the well known Doubly-Nonnegative (DNN) relaxation to approximate the completely positive and copositive programming problems. Specifically,

$$\begin{aligned} M \succcurlyeq_{cp} 0 &\approx M \succcurlyeq 0, M \geq 0 \\ M \succcurlyeq_{cop} 0 &\approx \exists M_1 \succcurlyeq 0, M_2 \geq 0, \text{ such that } M = M_1 + M_2. \end{aligned}$$

Note that we only need to obtain the most critical arc to be deleted from the network in each iteration of our dual-variable-based heuristic. Hence we require only the "ranking" of the importance of each arc, and not the precise solution value to the distributionally robust solution. Our computational results indeed show that the DNN relaxation can efficiently identify the most important arc in the network, with much less computational effort.

Let $\mathcal{G}^k$ represent the network configuration in deleting iteration k. We present the dual-variable-based heuristic in Table 2.

Table 2 Dual-Variable-Based Heuristic.

Step 1. Set the initial configuration to be a full flexible graph, i.e., $\mathcal{G}^0 = \mathcal{G}^F$. Set $k = 0$.
Step 2. Solve the DNN relaxation of copositive (COP) program (11) under configuration $\mathcal{G}^k$. Obtain the dual variable Γ^* corresponding to the constraint set $\mathbf{h}_i^\intercal X \hat{\mathbf{h}}_j + (f_i \hat{\mathbf{h}}_j^\intercal + \hat{f}_j \mathbf{h}_i^\intercal)\mathbf{p} + f_i \hat{f}_j = 0, \forall (i,j) \in \mathcal{G}^k$.
Step 3. Select the element Γ^*_{ab} with the smallest absolute value subject to $(a,b) \in \mathcal{G}^k$. If there are multiple elements of the smallest absolute value, randomly select one.
Step 4. Update the configuration to $\mathcal{G}^{k+1} = \mathcal{G}^k \backslash (a,b)$.
Step 5. Stop if the configuration obtained—i.e., $\mathcal{G}^{k+1}$—has the desired number of arcs. Otherwise, $k \leftarrow k+1$ and go to step 2.

Remark: Our heuristic is to remove arcs from a fully flexible structure. Alternatively, the efficient sparse structure can be built from scratch by greedily adding the most effective arc into the structure through solving one DNN relaxation of the copositive program for each arc that can be added. In this approach, we have to solve $O(|\mathcal{G}^0|)$ number of DNN relaxations in each iteration, which requires solving close to $k|\mathcal{G}^0|$ conic programs to get a k-arcs structure. In contrast, our dual-variable-based heuristic is computationally more attractive, as it only requires the solutions to $|\mathcal{G}^0|-k$ conic programs.

4. Applications

In this section, we apply the method derived in Section 3 to the roving team deployment and the process flexibility problem. Both can be modeled under the general stochastic network flow framework proposed in Section 3.

4.1. The Roving Team Deployment Problem

Demand for a roving team depends on various uncertain factors, such as passenger load for each flight, arrival bunching patterns, etc, and only occurs when the queue is sufficiently long. Demand for a roving team at each gate is therefore a random variable. To simplify the problem, we assume that once a roving team is required, it will remain for the entire duration—i.e., from the time the gate opens to the time the flight departs.[5].

We model the roving team deployment problem as a min-cost network flow problem. The network is built in "time" and "space" dimensions. Each node is identified by its gate coordinate and gate opening time. Let ts_i denote the time the gate of node i opens and te_i denote the time the flight departs from the gate. The duration from gate open ts_i to flight departure te_i is called the "time window" of node i. Let tr_{ij} denote the time needed to travel from the gate represented by node i to the gate represented by node j. An arc is connected from node i to node j if $te_i + tr_{ij} \leq ts_j$. Denote the constructed network as $\mathcal{G}(\mathcal{V}_0, \mathcal{A}_0)$, with $\mathcal{V}_0$ denoting the node set and $\mathcal{A}_0$ denoting the arc set. The number of nodes in the network is n, i.e., $|\mathcal{V}_0| = n$. To complete the network, we add two virtual nodes—one source node s and one sink node t—and s, t are linked to every node in $\mathcal{V}_0$. We denote the complete network as $\mathcal{G}(\mathcal{V}, \mathcal{A})$.

Before exploring the sparse structure, we first define "fully flexible structure" and "dedicated structure" . Note that an arc can be set up from node i to node j only when $te_i + tr_{ij} \leq ts_j$. We connect all the pair-wise arcs that satisfy these constraints to form a fully flexible structure denoted as $\mathcal{G}^F$. The dedicated structure, $\mathcal{G}^D$, is constructed in following way: Given all the possible arcs (those in the fully flexible structure), assuming all of the nodes need a roving team (i.e., demand $d_i = 1$ for all i), we solve the linear programming problem (13) in order to find the minimum total number of roving teams needed. The resulting routes generated by this linear programming gives the dedicated structure.

[5] Alternatively, we can partition the entire time window into multiple segments, and duplicate the node as many times as needed to determine demand for the roving team in each segment of the time window.

We denote the random demand for a roving team in each node $i \in \mathcal{V}_0$ as $\tilde{d}_i$. $\tilde{d}_i = 1$ indicates that node i needs a roving team, whereas $\tilde{d}_i = 0$ means that it does not. We model the deployment problem as a min-cost network flow problem as follows.

$$
\begin{array}{llll}
Z(\mathbf{d}) = \min & \sum\limits_{i \in \mathcal{V}_0, (s,i) \in \mathcal{A}} x_{si} & & \\
s.t. & \sum\limits_{i \in \mathcal{V}_0 \bigcup \{s\}, (i,j) \in \mathcal{A}} x_{ij} & \geq d_j, & j \in \mathcal{V}_0 \\
& \sum\limits_{i \in \mathcal{V}_0 \bigcup \{s\}, (i,j) \in \mathcal{A}} x_{ij} - \sum\limits_{i \in \mathcal{V}_0 \bigcup \{t\} (j,i) \in \mathcal{A}} x_{ji} & = 0, & j \in \mathcal{V}_0 \\
& x_{ij} & \geq 0 &
\end{array}
\tag{13}
$$

It is clear that (13) fits the general stochastic network problem proposed in Section 3 and satisfies the special network structure and cost structure required by (5). Therefore, we can apply quadratic reformulation to reduce the problem dimension. By replacing $\mathbf{y}$ with $\mathbf{1} - \mathbf{y}$, we can equivalently solve (13) as follows:

$$
\begin{array}{llll}
Z(\tilde{\mathbf{d}}) = \max\limits_{\mathbf{y},\mathbf{z}} & \sum\limits_{j \in \mathcal{V}_0} \tilde{d}_j (1 - y_j) & & \\
s.t & y_j + s_j^{(1)} = & 1, & j \in \mathcal{V}_0 \\
& z_j + s_j^{(2)} = & 1, & j \in \mathcal{V}_0 \\
& -y_j + z_j + s_j = & 0, & j \in \mathcal{V}_0 \\
& (1 - y_j + z_j)(1 - z_i) = & 0, & (i,j) \in \mathcal{A}_0 \\
& \mathbf{y}, \mathbf{z} \in & \{0,1\}^n, &
\end{array}
\tag{14}
$$

where two sets of valid constraints $\mathbf{y} + \mathbf{s}^{(1)} = \mathbf{1}$ and $\mathbf{z} + \mathbf{s}^{(2)} = \mathbf{1}$ are added.

The constraints in (14) satisfy Assumptions A1 to A3 in Section 3.1. We can construct an equivalent completely positive program following a similar procedure in Section 3.1. Note that the random coefficients in this problem are Bernoulli random variables. We impose the moment constraints defined by (10) in the completely positive program to ensure that the random variables constructed are indeed Bernoulli random variables. We refer the readers to the detailed formulation of the completely positive program as well as its dual formula copositive program in Appendix B.

In order to apply the dual-variable-based heuristic to design a sparse structure, we first establish strong duality property in Proposition 3.

Given a network, the Reachability Matrix $R \in \mathbb{R}^{n \times n}$ is defined as follows:

Definition 1. $R_{ij} = 1$ if there exists a path from i to j in $\mathcal{G}(\mathcal{V}_0, \mathcal{A}_0)$[6], and $R_{ii} = 1, i = 1, ..., n$.

Let J denote the matrix in $\mathbb{R}^{n \times n}$ with all the entries equal to 1.

Proposition 3. *Under the condition that (1) there exists a Bernoulli distribution satisfying the moment constraints and with the support containing a basis of $\mathbb{R}^n$ and having a nonempty*

[6] Note that the Reachability Matrix R can be obtained from the Adjacent matrix A, which is defined as $A_{ij} = 1$, if $(i,j) \in \mathcal{A}_0$, otherwise $A_{ij} = 0$. Define an operator $\star$ such that $A \star A$ returns a matrix denoted as B, and $B_{ij} = 1$ if $\sum\limits_{l=1}^{n} A_{il} A_{lj} > 0$, otherwise $B_{ij} = 0$. Denote $A^{[k]} \star A := A^{[k+1]}, \forall k \geq 1$. Then $R = \sum\limits_{i=1}^{n} A^{[i]}$.

intersection with positive orthant; and (2) $J - R^\intercal$ is nonsingular, then strong duality holds between a completely positive program and a copositive program that corresponds to the worst-case expected value of (14).

In the roving team deployment problem, Condition (1) holds by construction, since the moment conditions are generated from a demand distribution which satisfies Condition (1).

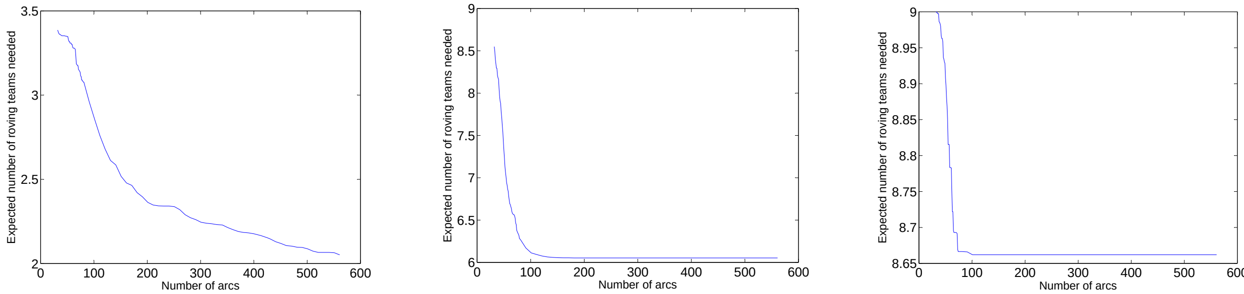
Building on the strong duality result, we have that the increase in the worst-case expected value of (14) after deleting arc (a,b) is bounded by $\frac{5}{4}|\Gamma^*_{ab}|$ according to Proposition 2. In the following, we will apply the dual-variable-based heuristic to design a sparse structure in the roving team deployment problem.

4.1.1. Implementation We use the Pre-board Deployment Roster Report and the gate layout of Changi Airport to build the network. More specifically, we build our network using the flight departure data from 10 am to 8 pm on April 4, 2014, for Terminal 1. We build the fully flexible structure by connecting each pair of nodes i and j, provided $te_i + tr_{ij} \leq ts_j$. In the numerical example presented in this section, we have a fully flexible graph with 40 nodes and 561 arcs. To get the dedicated structure, we solve the linear program (13) over fully flexible graph $\mathcal{G}^F(561)$ under the assumption that all the nodes need a roving team, i.e., $\mathbf{d} = \mathbf{1}_n$. The solution provides 9 routes from s to t, each of which does not intersect with every other route in $\mathcal{V}_0$. The resulting graph has 31 arcs. We denote the obtained dedicated structure as $\mathcal{G}^D(31)$.

Notice that the random demand variable in the roving team deployment problem is Bernoulli. The input of the moment matrix should be a valid moment matrix under a Bernoulli distribution family, which can be modelled by (10). As we only need the ranking of the dual variables, in this numerical implementation we did not incorporate the slack variables into the moment matrix, but simply enforce the constraints that the random variables are 0-1 (i.e. first moment equals to the second moment). Furthermore, to facilitate the comparison between performance and sparsity of the structure used, we assume the demand at each gate is independently and identically distributed. We compare three cases with demand probability for a roving team to be $p = 0.1$, $p = 0.5$, and $p = 0.9$.

Starting from the fully flexible structure $\mathcal{G}^F(561)$ built above, we use Mosek to solve the DNN relaxation of the copositive program (11) to get the dual variable Γ^*. We delete the arc (a,b) with $|\Gamma^*_{a,b}| = \min_{(i,j)\in\mathcal{G}^F(561),(i,j)\notin\mathcal{G}^D} \{|\Gamma_{ij}|\}$ to get a reduced graph, denoted as $\mathcal{G}^R$. Repeating this procedure in each iteration, we delete the arc (a,b) with $|\Gamma^*_{a,b}| = \min_{(i,j)\in\mathcal{G}^R,(i,j)\notin\mathcal{G}^D} \{|\Gamma_{ij}|\}$ and then update $\mathcal{G}^R \leftarrow \mathcal{G}^R \setminus (a,b)$. The procedure stops if the number of arcs in the graph reduces to 31—i.e., the dedicated graph structure $\mathcal{G}^D$.

To evaluate the performance of the sparse structure we obtained in the three cases ($p = 0.1$, $p = 0.5$, and $p = 0.9$), we use simulation to get the expected number of roving teams needed under



(a) prob$=0.1$ (b) prob$=0.5$ (c) prob$=0.9$

Figure 6 Expected number of roving teams needed under each sparse structure obtained using the dual-variable-based heuristic

each sparse structure. The simulation is conducted as follows: We first uniformly generate 2,000 samples with the prescribed demand distribution, and under each sparse structure $\mathcal{G}^R$ we solve one linear program (13) for each sample. Each linear program (13) gives the minimum number of roving teams needed under the specific demand scenario. We take the average to get the expected minimum number of roving teams required under the sparse structure $\mathcal{G}^R$ obtained from the heuristic.

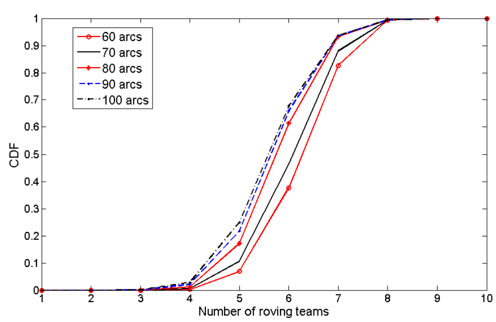
The expected number of roving teams needed under each sparse structure is shown in Figure 6. We observe that under the assumption of independent and identically distributed demand, to attain a performance level close to the fully flexible structure, an environment with higher (lower) likelihood of demanding a roving team requires less (more) flexibility. To see this more clearly, we focus on a specific sparse structure obtained from the dual-variable-based heuristic: the 62-arc structure, denoted as $\mathcal{G}^R(62)$. We select this sparse structure because it has twice the number of arcs as in the dedicated graph, $\mathcal{G}^D(31)$. We compare expected performance (expected number of roving teams) of the dedicated graph, $\mathcal{G}^D(31)$; the 62-arc structure, $\mathcal{G}^R(62)$; and the full graph, $\mathcal{G}^F(561)$. Results are shown in Table 3. We see that when demand likelihood is high—e.g., $p=0.9$—$\mathcal{G}^R(62)$ is almost as good as $\mathcal{G}^F(561)$. Actually, in this case all three structures perform almost the same. On the other hand, when demand likelihood is low—for example, in the case $p=0.1$—$\mathcal{G}^R(62)$ performs more like the dedicated graph $\mathcal{G}^D(31)$, and both perform much worse than the fully flexible structure $\mathcal{G}^F(561)$. In this case, we need more flexibility in the system.

Table 3 Expected number of roving teams under different graph structures

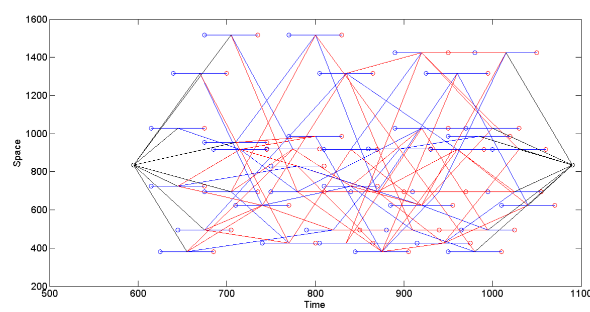
	$\mathcal{G}^F(561)$	$\mathcal{G}^R(62)$	$\mathcal{G}^D(31)$
prob =0.1	2.0515	3.2790	3.4025
prob =0.5	6.0525	6.6825	8.5760
prob =0.9	8.6620	8.7510	9

Interestingly, when $p=0.5$, the performance curve (Figure 6b) shows that numerous arcs (more than 400) can be deleted without sacrificing system performance. However, when the number of arcs is reduced below a certain threshold (e.g. slightly fewer than 100), any further reduction in flexibility becomes costly and system performance quickly deteriorates. This phase-transition phenomenon renders the flexibility design problem in this setting particularly challenging and important in practice.

We also plot the performance of various structures when $p = 0.5$ in our problem. The number of arcs in our structures vary between 60 to 100. Note that the 100-arc structure already performs as well as the fully flexible structure. We plot the empirical cumulative distribution functions of the number of roving teams required to fulfill the demand under simulation in Figure 7a. From the figure we can see that although the system requires 9 roving teams in the worst case, when $p = 0.5$, we can use 8 teams to serve the demand with a 60-arc structure almost all the time. On the other hand, using 7 teams on an 80-arc structure (c.f. Figure 7b), we can cover the demand more than 95% of the time. Increasing the number of arcs further to 90 or 100 arcs has negligible impact on performance.



(a) Sample path performance under different structures(Prob=0.5)

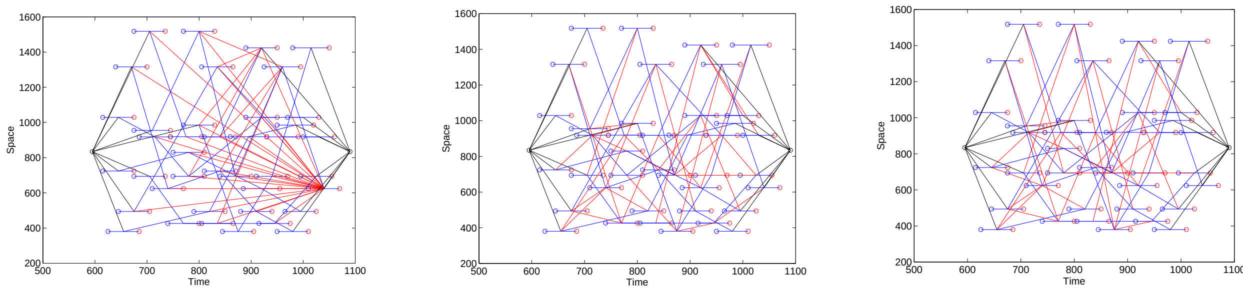


(b) Efficient deployment structure with 7 teams available

We next evaluate the performance of the 62-arc structure obtained from the dual-variable-based heuristic, under three different demand scenarios - $p = 0.1$, $p = 0.5$, and $p = 0.9$ respectively (cf. Figure 7). The 62−arc structures obtained for the cases when $p = 0.5$ and $p = 0.9$ are quite similar, with many identical arcs (they only differ in the choice of 5 arcs). On the other hand, as shown in Table 3 (second and third rows), these two 62-arc structures both perform well when $p = 0.5$ and $p = 0.9$. This indicates that the proposed heuristic does indeed help to select the more effective arcs for roving team deployment.

4.2. The Process Flexibility Problem

One of the central problems studied in process flexibility literature is to design a sparse structure that links plant nodes and product nodes such that demand for products can be better fulfilled. Specifically, a k-chain structure is a well-known efficient structure when the system is balanced



(c) prob$= 0.1$ (d) prob$= 0.5$ (e) prob$= 0.9$

Figure 7 62−arc structure under different demand probabilities

and symmetric[7]. In this section, we will apply the dual-variable-based heuristic to design efficient sparse structures for the process flexility problem.

Suppose we are given a bipartite graph, with the set of plant nodes, $\mathcal{I}$ ($|\mathcal{I}| = n$); the set of product nodes, $\mathcal{J}$ ($|\mathcal{J}| = m$); and arcs that connect plant nodes and product nodes, $\mathcal{A}_0 = \{(i,j) \mid i \in \mathcal{I}, j \in \mathcal{J}\}$. Denote the total set of nodes $\mathcal{I} \cup \mathcal{J}$ as $\mathcal{V}_0$. We add a virtual source node s and virtual sink node t, with s and t linking to every node in $\mathcal{V}_0$. Let $\mathcal{A}$ be the set of arcs in the new graph after adding s and t.

Consider the following analogous model studied in Section 3 :

$$
\begin{aligned}
Z(\tilde{\mathbf{d}}) = \max_{x_{ij}} \quad & \sum_{(i,j)\in\mathcal{A}_0} x_{ij} & & \\
s.t. \quad & \sum_{i\in\mathcal{V}_0 \bigcup \{s\},(i,j)\in\mathcal{A}} x_{ij} \le \tilde{d}_j, & & j \in \mathcal{I}\cup\mathcal{J} \\
\sum_{i\in\mathcal{V}_0 \bigcup \{s\},(i,j)\in\mathcal{A}} x_{ij} - & \sum_{i\in\mathcal{V}_0 \bigcup \{t\}(j,i)\in\mathcal{A}} x_{ji} = 0, & & j \in \mathcal{I}\cup\mathcal{J} \\
& x_{ij} \ge 0, & &
\end{aligned}
\tag{15}
$$

where the support of $\tilde{\mathbf{d}}$ is nonnegative. Take the dual of Problem (15).

$$
\begin{aligned}
Z(\tilde{\mathbf{d}}) = \min_{\mathbf{y},\mathbf{z}} \quad & \sum_{j\in\mathcal{I}\cup\mathcal{J}} \tilde{d}_j y_j & \\
s.t. \quad & y_j + z_j \ge 0, & j \in \mathcal{I}\cup\mathcal{J} \\
& y_j + z_j - z_i \ge 1, & (i,j) \in \mathcal{A}_0 \\
& -z_i \ge 0, & i \in \mathcal{I}\cup\mathcal{J} \\
& \mathbf{y} \ge \mathbf{0}. &
\end{aligned}
\tag{16}
$$

Due to total unimodularity and the special nature of the bipartite graph, we can further simplify the model before reformulating it in quadratic form. Since the support of $\tilde{\mathbf{d}}$ is nonnegative, it can

[7] Suppose there are n plant nodes and n product nodes. A k-chain structure is a general symmetric graph, where plant 1 connects to product 1 to product k, plant 2 connects to product 2 to product $k+1$, and, in general, plant i connects to i; $i+1$;...; $i+k-1$ (modulo n). The dedicated graph and full flexibility graph are both special cases of a k-chain, with $k = 1$ and $k = n$

be verified that in the optimal solution, $z_i = -y_i, \forall i \in \mathcal{I}$, $z_j = 0, \forall j \in \mathcal{J}$. Hence we get the following formulation which is equivalent to (16):

$$\begin{aligned} Z(\tilde{\mathbf{d}}) = \min_{\mathbf{y}} & \sum_{j \in \mathcal{I} \cup \mathcal{J}} \tilde{d}_j y_j \\ s.t. \quad & y_j + y_i \geq 1, \quad (i,j) \in \mathcal{A}_0 \\ & \mathbf{y} \geq \mathbf{0}. \end{aligned} \tag{17}$$

With a slight abuse of notation, let $\mathbf{c} \in \mathbb{R}^n$ denote the capacity of n plants, which we assume is deterministic, and let $\tilde{\mathbf{d}}$ be the random variables that represent the random demands of m products. By refining $\mathbf{y}$ as $\begin{pmatrix} \mathbf{y} \\ \mathbf{z} \end{pmatrix}$, where $\mathbf{y} \in \mathbb{R}^m, \mathbf{z} \in \mathbb{R}^n$, we can get the following reformulation of the optimization problem defined by (17).

$$\begin{aligned} Z(\tilde{\mathbf{d}}) = \min & \sum_{i \in \mathcal{I}} c_i z_i + \sum_{j \in \mathcal{J}} \tilde{d}_j y_j \\ s.t. \quad y_j + z_i \geq & \; 1, \quad (i,j) \in \mathcal{A}_0 \\ y_j \geq & \; 0, \quad j \in \mathcal{J} \\ z_i \geq & \; 0, \quad i \in \mathcal{I} \end{aligned} \tag{18}$$

Interestingly, (18) is exactly the dual formulation to the max-flow problem in the bipartite graph, which is the model used in the literature to model the process flexibility problem. [8]

Problem (18) can also be reformulated as a quadratic constrained problem based on total unimodularity. We can equivalently replace $y_j + z_i \geq 1$ with $(1 - y_j)(1 - z_i) = 0$. After adding two sets of valid cuts $\mathbf{0} \leq \mathbf{y} \leq \mathbf{1}$, $\mathbf{0} \leq \mathbf{z} \leq \mathbf{1}$ and standardizing the constraints, we get an equivalent reformulation of (18) as follows:

$$\begin{aligned} Z(\mathbf{d}) = \max \quad & \sum_{i \in \mathcal{I}} -c_i z_i + \sum_{j \in \mathcal{J}} -d_j y_j \\ s.t. \quad & (1 - y_j)(1 - z_i) = 0, \quad (i,j) \in \mathcal{A}_0 \\ & \mathbf{y} + \mathbf{s_1} = \mathbf{1}, \\ & \mathbf{z} + \mathbf{s_2} = \mathbf{1}, \\ & \mathbf{y} \in \{0,1\}^m, \mathbf{z} \in \{0,1\}^n. \end{aligned} \tag{19}$$

This formulation satisfies Assumptions A1 to A3 in Section 3.1. Hence, we can obtain the completely positive program, which is equivalent to the worst-case max flow, $\sup_{\tilde{\mathbf{d}} \sim (\boldsymbol{\mu}_d, \Sigma_d)} \boldsymbol{E}[Z(\tilde{\mathbf{d}})]$, given the first-two moments of demand $\tilde{\mathbf{d}}$ as $\boldsymbol{\mu}_d$ and Σ_d. We then follow the construction of (11) to get the corresponding copositive program. For the detailed conic formulation, see Appendix B.

Additionally, it is interesting to observe that (19) is a variation of the classic stable set problem. Since $\mathbf{y} \in \{0,1\}^m$, $\mathbf{z} \in \{0,1\}^n$, we can treat $\hat{y}_j := (1 - y_j)$ and $\hat{z}_i := (1 - z_i)$ as indicator variables of

[8] The maximum flow used in modeling the process flexibility problem can be written as follows:

$$\begin{aligned} Z(\tilde{\mathbf{d}}) \quad & = \max \sum_{(i,j) \in \mathcal{A}_0} x_{ij} \\ s.t. \sum_{i \in \mathcal{I}, (i,j) \in \mathcal{A}_0} x_{ij} & \leq \tilde{d}_j, \quad j \in \mathcal{J} \\ \sum_{j \in \mathcal{J}, (i,j) \in \mathcal{A}_0} x_{ij} & \leq c_i, \quad i \in \mathcal{I} \\ x_{ij} & \geq 0 \end{aligned}$$

nodes j or i belonging to a stable set; e.g. S. The constraints, $(1-y_j)(1-z_i)=0,(i,j)\in\mathcal{A}_0$—or, in terms of $\hat{y}_j$ and $\hat{z}_i$, $\hat{y}_j\hat{z}_i=0,(i,j)\in\mathcal{A}_0$—imply that the two nodes connected by an arc in graph $\mathcal{G}(\mathcal{V},\mathcal{A})$ cannot be in the same stable set S. In the case in which $\tilde{\mathbf{d}}$ is random, the problem can be regarded as a random-weight stable set problem. As described in Section 2, de Klerk and Pasechnik (2002) solve the maximal stable set problem using a copositive program. In contrast, we show that the worst-case value of this random-weight stable set problem can also be equivalently reformulated as a copositive program.

Similar to the roving team deployment problem, we first establish strong duality property in the following proposition:

Proposition 4. *Under the condition that the moment matrix lies in the interior of completely positive cone $\mathcal{C}_m^*$, strong duality holds between a completely positive program and a copositive program corresponding to the worst-case expected value of (19).*

As mentioned above, through proper reformulation, the flexibility problem (16) is in fact a special case of the general network flow model presented in Section 3, with $c_{si}=1,c_{ij}=0,c_{it}=0$. According to Proposition 2, combined with strong duality in Proposition 4, the change in the worst-case expected value of the fulfilled demand is bounded by $\frac{5}{4}|\Gamma_{ab}^*|$ after deleting arc (a,b). We show by applying the simplified formulation (18), which is obtained by exploiting the special nature of the bipartite graph, the bound of objective change when deleting arc (a,b) can be improved to $|\Gamma_{ab}^*|$.

Proposition 5. *The increase in the worst-case expected value of (19) after deleting arc (a,b) is bounded above by the absolute value of the optimal dual solution $|\Gamma_{ab}^*|$.*

4.2.1. Implementation We use the heuristic to design efficient sparse process structure for three classes of problems in this area: when (1) the system is balanced and symmetric, (2) when the system is balanced but asymmetric, and (3) on the GM problem studied in Jordan and Graves (1995). The machine we use to perform the computation is Intel(R) Core(TM) i5-4690 CPU 3.50 GHz, RAM 8 GB, Microsoft Windows Windows 10. The solver is Enterprise cvx Mosek solver.

(1) The Symmetric and Balanced System

We first apply the proposed dual-variable-based heuristic to a symmetric and balanced system. A system is symmetric if all plants have the same capacity and all products have i.i.d. demand, with mean demand identical to plant capacity. The system is balanced if there are an equal number of products and plants.

In our numerical example, there are 5 products with mean demand of 30 and coefficient of variation of 0.4, and 5 plants with the same capacity of 30.

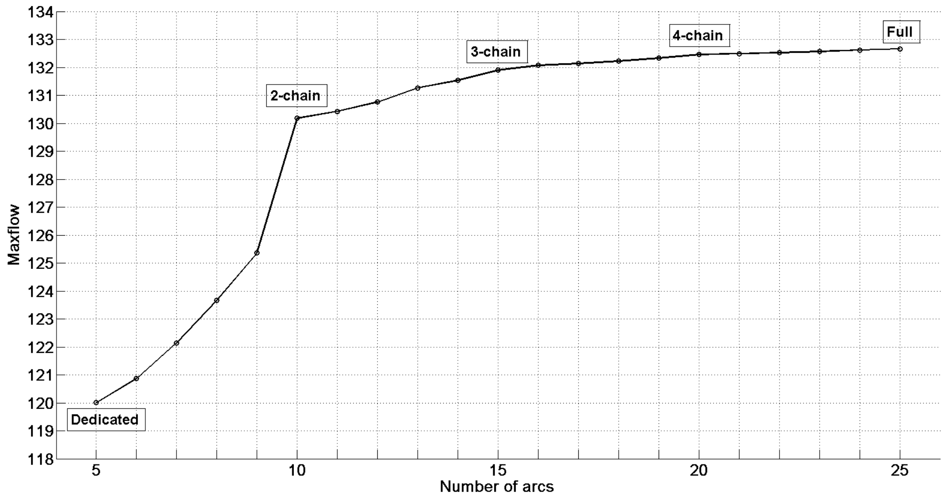


Figure 8 The worst-case expected performances (Balanced but symmetric case)

Starting with a full-flexibility structure, we apply the dual-variable-based heuristic to incrementally reduce the number of arcs from 25 to 5. Interestingly, the structures we obtain from the heuristic with 10, 15, and 20 arcs are exactly the 2-chain, 3-chain, and 4-chain structures. When the number of arcs reaches 5, we have the dedicated structure. The respective worst-case expected maximum flows are shown in Figure 8. Despite using a slightly weaker set of conditions on demand, our results mirror one well-known key insight in process flexibility literature, which is that the 2-chain performs significantly better than the dedicated structure and nearly as well as the fully flexible structure. This insight holds for different symmetric and balanced process flexibility systems with different numbers of products and plants. The largest system we analyzed has 27 products and 27 plants, and our heuristic continues to recover all k-chain structures.

Our success in recovering k-chains provides partial evidence that the dual-variable-based heuristic approach is able to construct a good sparse process structure for this type of problem, even though it may not return the optimal process structure[9].

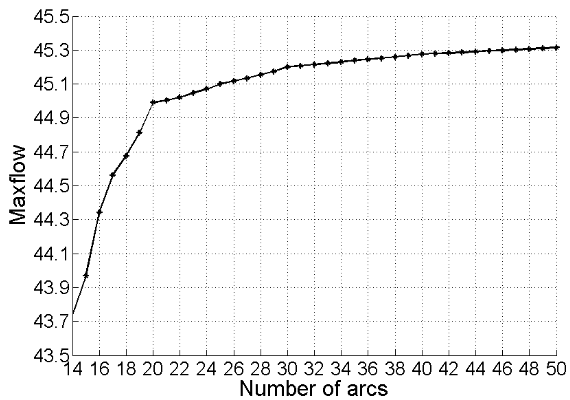
(2) The Unbalanced System

We consider next unbalanced systems. Specifically, we consider the cases where all plants have the same capacity and all products have i.i.d. demand. The total capacity equals to the total mean demand. However, the total number of plants is not the same as the total number of products.

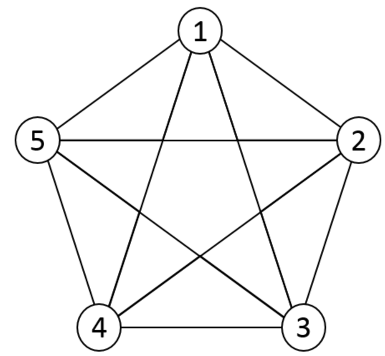
[9] We have also implemented a heuristic to insert arc greedily, using the value of the conic programming problem as a gauge of the value of an arc, starting from a null graph. Unfortunately this more natural greedy heuristic does not return the 2-chain after adding $2n$ arcs. In fact, the structure obtained may not even be connected.

Example 1. There are 5 plants each with capacity of 10, and 10 products each with mean demand of 5. The coefficient of variation of demand distribution is 0.4.

We apply the dual-variable-based heuristic to incrementally reduce the number of arcs from 50 to 14.



(a) The worst-case expected performances (Unbalanced but symmetric example 1)



(b) 20-arc structure

The worst-case expected maximum flows are plotted in Figure 9a. We observe there is a dramatic drop in performance when the number of arcs goes below 20, but improves marginally when the number of arcs goes above 20. The worst-case performance of the structure with 20 arcs is very close to the performance under full flexibility graph (gap of 0.6%). More interestingly, the 20-arc structure obtained is symmetric, with each plant connected to 4 demand nodes, and each demand node connected to 2 plants. To visualize this structure, we construct a graph by linking all the plant nodes that are connected to the same product node with an arc. The graph (which we call “20 arc” structure) obtained is shown in Figure 9b. It is a 5-node complete graph.

Example 2. There are 10 plants each with capacity of 3 and 15 products each with mean demand of 2. The coefficient of variation of the demand distribution is 0.4.

By implementing the dual-variable-based heuristic to incrementally reduce the number of arcs from 150 to 15, we have the worst-case expected maximum flows in Figure 9. We find that the 30-arc structure obtained is a very efficient and sparse structure. Furthermore, the performance drops dramatically for structures with less than 30 arcs, but improves only marginally for structures with more than 30 arcs. This 30-arc structure is also symmetric. Each plant is connected to 3 products and each product node connected to 2 plants. To visualize this structure, we construct a graph by linking all the plant nodes which are connected to the same product node with an arc (“30-arc

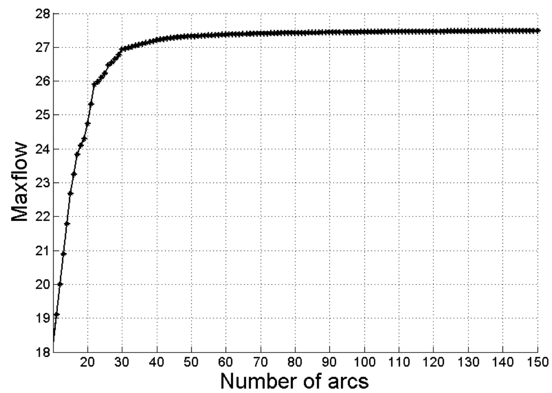
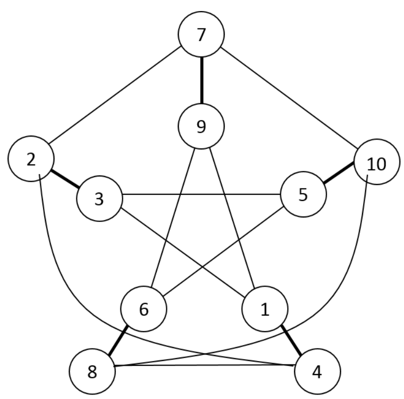


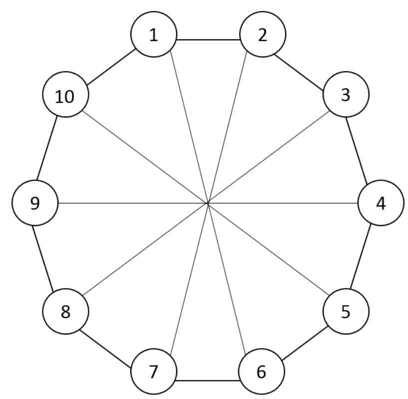
Figure 9 **The worst-case expected performances (Unbalanced but symmetric example 2)**

A" in Figure 10a). Interestingly, this graph is very similar to the Petersen Graph which is a well known non-hamiltonian graph.

Deng and Sheng (2013) conjectured that the optimal structure for this numerical example is a 2 chain on 10 nodes with 5 additional diagonal arcs ("30-arc DS" in Figure 10b). Interestingly, in



(a) 30-arc A



(b) 30-arc DS

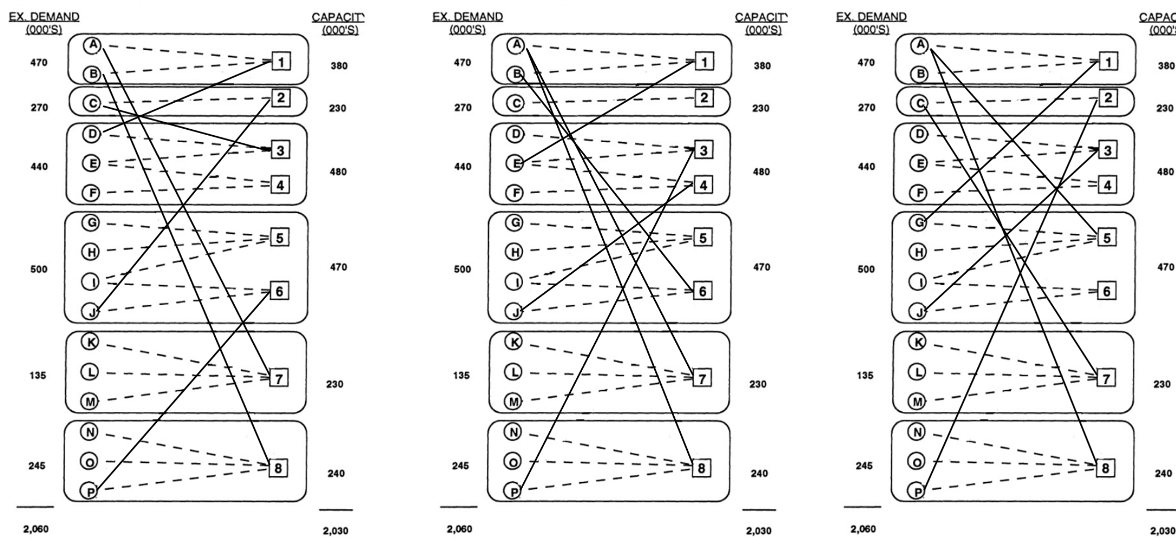
numerical simulation, the two structures have almost identical performance on average, with our structure having a slight edge in terms of worst-case and sample path performance. (cf. Table 4).

(3) General System

Jordan and Graves (1995) used data from an automobile manufacturer and constructed an efficient sparse structure. Demand means and plant capacities are given in Figure 10c. They assumed that demands are truncated ($\pm 2\sigma$) normally distributed. The coefficient of variation for each product is 0.4. Products fall into 3 groups: A to F, G to M, and N to P. They assumed that intergroup pairs have 0 correlation, and pairs of products within the same group are correlated with a coefficient of 0.3. They used simulation to obtain an efficient 6-arc structure, which achieved almost

Table 4 Numerical example 2 structure comparison

Worst-case Scenario Comparison		
Configuration	Worst Case Expected Sales (Thousand units)	Capacity Utilization
30-arc A	26.94	89.80%
30-arc DS	26.93	89.76%
Expected-case Performance Comparison		
Configuration	Expected Sales	95% Confidence Interval
30-arc A	28.893	[28.882, 28.903]
30-arc DS	28.893	[28.882, 28.903]
Sample-path Performance Comparison		
Configuration	30-arc A Outperform Percentage	30-arc DS Outperform Percentage
	50.17%	49.83%



(c) JG (d) A (e) B

Figure 10 Comparison of three 6-arc structures (Jordan and Graves (1995))

the same expected performance as the fully flexible configuration. This 6-arc structure is shown in Figure 10c.

In our numerical study, we do not assume knowledge of the full demand distribution. Instead, we only assume that the first and second moments are known and apply the dual-variable-based heuristic to obtain a 6-arc structure (structure A, Figure 10d). Since our approach studies the worst case, we first analyze the worst-case performance of Jordan and Graves's(1995) 6-arc structure (JG) to make a comparison. The total worst-case sales under their structure is 1755.669 thousand units. Our structure A attained performance (1750.102) that is worse off by only 0.32%.

To obtain better structures, we first identify the arc in the A structure with the smallest absolute dual solution (in this case, arc $e = (B, 6)$). We next re-run the dual-variable based heuristic on the fully flexible graph, with arc e deleted. The new structure obtained has worst-case expected sales 1751.182. It is better than the performance of structure A, but is still worse than structure JG.

Repeating this procedure one more time, we obtained structure B (in Figure 10e), which performs better than JG in the worst case (the worst-case expected sales is 1759.488).

It is interesting that by repeating this procedure only twice, we already obtain a structure better than JG. Our heuristic takes CPU time 211s, or around 3 to 4 minutes to execute. This is conceivably faster than the trial-and-error methods used to construct JG.

We compare next the performance of the three structures using simulation, assuming that the demands follow multivariate normal distributions with the given parameters. We generate $50,000$ demand samples using Jordan and Graves's(1995) demand distribution specification, to get the expected performance for each of the three structures. The expected performances of structure B and JG are almost identical. This confirms our belief that a structure that performs well in the worst case will also perform well on average, at least when the demands follow a normal distribution. From the perspective of sample-path performance, the structure B outperforms JG slightly, beating it 50.2% to 49.8%. All performance comparison results are shown in Table 5.

Table 5 Structure Comparison

Worst-case Scenario Comparison		
Configuration	Worst Case Expected Sales (Thousand units)	Capacity Utilization
6-arc JG	1755.669	86.48%
6-arc A	1750.102	86.21%
6-arc B	1759.488	86.67%
Expected-case Performance Comparison		
Configuration	Expected Sales (Thousand units)	95% Confidence Interval
6-arc JG	1930.799	[1927.786, 1933.812]
6-arc A	1906.024	[1902.942, 1909.106]
6-arc B	1930.799	[1927.786, 1933.812]
Sample-path Performance Comparison		
Configuration	6-arc JG Outperform Percentage	6-arc A(B) Outperform Percentage
6-arc JG vs. 6-arc A	71.87%	28.13%
6-arc JG vs. 6-arc B	49.80%	50.20%

5. Concluding Remark

We propose a novel approach to design sparse and efficient structures for operational problems. Our dual-variable-based heuristic constructs an efficient sparse structure by incrementally reducing the number of arcs in a network, based on sensitivity analysis of a related completely positive program, which is a distributionally robust reformulation of the worst-case network flow problem. We then apply this heuristic to two canonical problems in operations. It is interesting to see that the heuristic recovers the k-chain structure, which is a well-known efficient structure in the process flexibility literature. Moreover, it can be used to identify a well-performed sparse structure for general process flexility problems with asymmetric capacity and demand. We also apply the dual-variable-based heuristic to a roving team deployment problem and obtain an efficient sparse deployment network, and show that we can pinpoint the phase-transition threshold for the flexibility structure in this network.

Our approach exploits a distributionally robust reformulation for several classes of network flow problems into completely positive programs, and can potentially be extended to more general flexibility design problems (for a review of other classes of flexibility design problems, see Chou et al. (2008)). Our key insight is to reformulate an inequality constraint in a stochastic optimization problem using a quadratic equality constraint, so that the natural dual variable in an associated copositive cone can be viewed as a dual price of the inequality constraint in the stochastic optimization problem. This insight is also potentially useful for sensitivity analysis studies of other classes of stochastic optimization problems. We leave these and other issues for future research.

Acknowledgments

We thank for Sun De-feng and Toh Kim Chuan to provide their software SDPNAL+ to solve large scale Doubly Nonegative Problem in this project.

References

Ahuja R. K., Magnanti T. L., Orlin J.B., Reddy. 1995. Applications of Network Optimization. *Handbooks in OR & MS* **7** 24–27.

Bard J.F., Morton D.P., Wang Y.M. 2007. Workforce planning at USPS mail processing and distribution centers using stochastic optimization. *Annals of Operations Research* **155** 51–78.

Bergh J.V., Belin J., Bruecker P.D., Demeulemeester E., Boeck L.D. 2013. Personnel scheduling: A literature review. *European Journal of Operational Research* **226** 367–385.

Berman A., Shaked-Monderer N. 2003. Completely Positive Matrices. *World Scientific, Singapore, Republic of Singapore*

Bureau of Labor Statistics. 2013. The employment situationFebruary. *Retrieved March 17, 2013, http://www.bls.gov/news.release/pdf/empsit.pdf..*

Burer S. 2009. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming* **120** 479–495.

Chou M., Chua G. A., Teo C.P., Zheng H. 2010. Design for process fexibility: Effciency of the long chain and sparse structure. *Operations Research* **58**(1) 43–58.

Chou M., Chua G. A, Teo C.P., Zheng H. 2011. Process flexibility revisited: The graph expander and its applications. *Operations Research* **59**(5) 1090–1105.

Chou M., Teo C.P., Zheng H. 2008. Process flexibility: design, evaluation, and applications. *Flexible Services and Manufacturing Journal* **20**(1) 59–94.

Daniels R.L., Mazzola J.B., Shi D. 2004. Flow Shop Scheduling with Partial Resource Flexibility. *Management Science* **50**(5) 658–669.

de Klerk E., Pasechnik D.V. 2002. Approximation of the stability number of a graph via copositive programming. *SIAM Journal on Optimization* **12** 875–892.

Deng T.H., Shen Z.J. 2013. Process flexibility design in unbalanced networks. *Manufacturing Service Operations Management* **15**(1) 24–32.

Desir A., Goyal V., Wei Y.H., Zhang J.W. 2016. Sparse Process Flexibility Designs: Is Long Chain Really Optimal? *Operations Research* **64**(2) 416–431.

Dickinson P. 2010. An improved characterisation of the interior of the completely positive cone. *Electronic Journal of Linear Algebra* **20** 723–729.

Gaddum J.W. 1958. Linear inequalities and quadratic forms. *Pacific Journal of Mathematics* **8** 411–414.

Iravani S. M., Van Oyen M. P., Mark, Sims K. T. 2005. Structural flexibility: A new perspective on the design of manufacturing and service operations. *Management Science* **51**(2) 155–166.

Jordan W. C., Graves S. C. 1995. Principles on the Benefits of Manufacturing Process Flexibility. *Management Science* **41**(4) 577–594.

Kesavan S., Staats B.R., Gilland W. 2014. Volume Flexibility in Services: The Costs and Benefits of Flexible Labor Resources. *Management Science* **60**(8) 1884–1906.

Koffman D. 2004. Operational Experiences with Flexible Transit Services. *Transportation Research Board*

Kong Q. X., Lee C. Y., Teo C. P., Zheng Z. C. 2013. Scheduling Arrivals to a Stochastic Service Delivery System Using Copositive Cones. *Operations Research* **61**3 711–726.

Li X.B., Natarajan K., Teo C.P., Zheng Z.C. 2014. Distributionally Robust Mixed Integer Linear Programs: Persistency Models with Applications. *European Journal of Operational Research* **233**(3) 459–473.

Lovasz L. and Schrijver A., 1991. Cones of matrices and set-functions and 0-1 optimization. *SIAM journal on optimization* **1** 166–190.

Murty K.G., Kabadi S.N. 1987. Some NP-complete problems in quadratic and nonlinear programming *Math Programming* **39** 117–129.

Natarajan K., Teo C. P., Zheng Z. 2011. Mixed zero-one linear programs under objective uncertainty: a completely positive representation. *Operations Research* **59** 713–728.

Natarajan. K., Teo C.P. 2016. On Reduced Semidefinite Programs for Second Order Moment Bounds with Applications. *Math Programming.* Forthcoming.

Nobert Y. and Roy J. 1998. Freight handling personnel scheduling at air cargo terminals. *Transportation Science* **32**(3) 295–301.

Parrilo P. 2000. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. *Ph.D. Dissertation. California Institute of Technology*

Qin R., Nembhard D.A., Barnes W.L. 2015. Workforce flexibility in operations management. *Surveys in Operations Research and Management Science* **20**(1) 19–33.

Rockafellar R. T. 1970. Convex Analysis. *Princeton University Press*

Simchi-Levi D., Wei Y.H. 2012. Understanding the performance of the long chain and sparse designs. *Operations Research* **60**(5) 1125–1141.

Simchi-Levi D., Wei Y.H. 2015. Worst-case analysis of process flexibility designs. *Operations Research* **63**(1) 166–185.

Sun D.F., Toh K.C., Yang L.Q. 2015. A convergent 3-block semi-proximal alternating direction method of multipliers for conic programming with 4-type of constraints. *SIAM Journal on Optimization* **25**(2) 882–915.

Wang X., Zhang J.W. 2015. Process flexibility: A distribution-free bound on the performance of k-chain. *Operations Research* **63**(3) 555–571.

Yang L.Q., Sun D.F., Toh K.C. 2014. SDPNAL+: A majorized semismooth newton-cg augmented lagrangian method for semidefinite programming with nonnegative constraints. *Mathematical Programming Computation* **7**(3) 331–366.

Zhu X., Sherali H.D. 2009. Two-Stage Workforce Planning under Demand Fluctuations and Uncertainty. *The Journal of the Operational Research Society* **60**(1) 94–103.

Appendix A: Proofs

To prove Proposition 1, we first explore some properties of the feasible solutions to Z_C in the following two lemmas.

LEMMA 1. *(Natarajan et al. (2011), Proposition 3.1* & *Proposition* 3.2*): Let* $(\mathbf{p}, X)$ *be a feasible solution to* Z_C*, and consider any completely positive decomposition of matrix*

$$\begin{pmatrix} 1 & \mathbf{p}^\mathsf{T} \\ \mathbf{p} & X \end{pmatrix} = \sum_{k\in\kappa} \begin{pmatrix} \alpha_k \\ \boldsymbol{\gamma}_k \end{pmatrix} \begin{pmatrix} \alpha_k \\ \boldsymbol{\gamma}_k \end{pmatrix}^\mathsf{T} \tag{20}$$

where κ *is a set of finite indices such that* $\kappa = \kappa_+ \cup \kappa_0$*, where* $\kappa_+ = \{k \in \kappa \mid \alpha_k > 0\}$*, and* $\kappa_0 = \{k \in \kappa \mid \alpha_k = 0\}$*;* $\alpha_k \in \mathbb{R}_+$*,* $\sum_{k\in\kappa_+} \alpha_k^2 = 1$*,* $\boldsymbol{\gamma}_k \in \mathbb{R}_+^N, \forall k \in \kappa$*. Then*

1. $\mathbf{a}_i^\mathsf{T} \frac{\boldsymbol{\gamma}_k}{\alpha_k} = b_i, \forall i, \forall k \in \kappa_+$*;*
2. $\frac{\gamma_{kj}}{\alpha_k} \in \{0,1\}, \forall j \in \mathcal{B}, \forall k \in \kappa_+$*;*
3. $\boldsymbol{\gamma}_k = \mathbf{0}, \forall k \in \kappa_0$*.*

Lemma 1 confirms that each decomposition in κ_+ satisfies all the constraints in problem (7) except for the quadratic ones, $(\mathbf{h}_i^\mathsf{T}\mathbf{x} + f_i)(\hat{\mathbf{h}}_j^\mathsf{T}\mathbf{x} + \hat{f}_j) = 0, \forall (i,j) \in \mathcal{H}$. We further demonstrate that this set of constraints also holds for each decomposition $k \in \kappa_+$ in Lemma 2.

LEMMA 2. *Let* $(\mathbf{p}, X)$ *be a feasible solution to* Z_C*, and consider decomposition (20), then* $(\mathbf{h}_i^\mathsf{T} \frac{\boldsymbol{\gamma}_k}{\alpha_k} + f_i)(\hat{\mathbf{h}}_j^\mathsf{T} \frac{\boldsymbol{\gamma}_k}{\alpha_k} + \hat{f}_j) = 0, \forall k \in \kappa_+$*.*

Proof: Since $(\mathbf{p}, X)$ is a feasible solution to Z_C, then $\mathbf{h}_i^\intercal X \hat{\mathbf{h}}_j + (f_i \hat{\mathbf{h}}_j^\intercal + \hat{f}_j \mathbf{h}_i^\intercal)\mathbf{p} + f_i \hat{f}_j = 0, \forall (i,j) \in \mathcal{H}$. Rewrite it based on decomposition (20) as follows,

$$\mathbf{h}_i^\intercal \sum_{k \in \kappa_+} \alpha_k^2 \frac{\gamma_k}{\alpha_k} \frac{\gamma_k^\intercal}{\alpha_k} \hat{\mathbf{h}}_j + (f_i \hat{\mathbf{h}}_j^\intercal + \hat{f}_j \mathbf{h}_i^\intercal) \sum_{k \in \kappa_+} \alpha_k^2 \frac{\gamma_k}{\alpha_k} + f_i \hat{f}_j = 0, \forall (i,j) \in \mathcal{H} \tag{21}$$

Note that $\sum_{k \in \kappa_+} \alpha_k^2 = 1$. Rearranging (21), we get

$$\sum_{k \in \kappa_+} \alpha_k^2 (\mathbf{h}_i^\intercal \frac{\gamma_k}{\alpha_k} + f_i)(\hat{\mathbf{h}}_j^\intercal \frac{\gamma_k}{\alpha_k} + \hat{f}_j) = 0 \tag{22}$$

From Lemma 1, we have $\mathbf{a}_i^\intercal \frac{\boldsymbol{\gamma}_\mathbf{k}}{\alpha_k} = b_i, \forall k \in \kappa_+$. According to Assumption 1, we have $\mathbf{h}_i^\intercal \frac{\gamma_k}{\alpha_k} + f_i \geq 0, \hat{\mathbf{h}}_j^\intercal \frac{\gamma_k}{\alpha_k} + \hat{f}_j \geq 0, \forall k \in \kappa_+$. Combined with (22), we then obtain $(\mathbf{h}_i^\intercal \frac{\gamma_k}{\alpha_k} + f_i)(\hat{\mathbf{h}}_j^\intercal \frac{\gamma_k}{\alpha_k} + \hat{f}_j) = 0, \forall k \in \kappa_+$. Q.E.D.

Now we are ready to prove Proposition 1.

Proof of Proposition 1 Let $(\mathbf{p}, X)$ be an optimal solution to Z_C. Consider a completely positive decomposition of the matrix

$$\begin{pmatrix} 1 & \boldsymbol{\mu}_d{}^\intercal & \mathbf{p}^\intercal \\ \boldsymbol{\mu}_d & \Sigma_d & Y^\intercal \\ \mathbf{p} & Y & X \end{pmatrix} = \sum_{k \in \kappa_+} \alpha_k^2 \begin{pmatrix} 1 \\ \frac{\boldsymbol{\beta}_k}{\alpha_k} \\ \frac{\boldsymbol{\gamma}_k}{\alpha_k} \end{pmatrix} \begin{pmatrix} 1 \\ \frac{\boldsymbol{\beta}_k}{\alpha_k} \\ \frac{\boldsymbol{\gamma}_k}{\alpha_k} \end{pmatrix}^\intercal + \sum_{k \in \kappa_0} \begin{pmatrix} 0 \\ \boldsymbol{\beta}_k \\ \boldsymbol{\gamma}_k \end{pmatrix} \begin{pmatrix} 0 \\ \boldsymbol{\beta}_k \\ \boldsymbol{\gamma}_k \end{pmatrix}^\intercal$$

According to Lemma 1 and Lemma 2 we have

$\forall k \in \kappa_+$, $\mathbf{a}_i^\intercal \frac{\gamma_k}{\alpha_k} = b_i, \forall i$; $\frac{\gamma_k}{\alpha_k} \in \{0,1\}^N$ and $(\mathbf{h}_i^\intercal \frac{\gamma_k}{\alpha_k} + f_i)(\hat{\mathbf{h}}_j^\intercal \frac{\gamma_k}{\alpha_k} + \hat{f}_j) = 0, \forall (i,j) \in \mathcal{H}$, and $\boldsymbol{\gamma}_k = \mathbf{0}, \forall k \in \kappa_0$.

In other words, each decomposition $\frac{\gamma_k}{\alpha_k}$ is a feasible solution to (4). To complete the proof, we use a similar argument as in Natarajan et al. (2011)—we construct a sequence of random vectors whose limit satisfies the moment condition and a corresponding sequence of feasible solutions. The limit of the set of feasible solutions under such a distribution sequence achieves the lower bound provided by Z_C. The details are omitted here, and we refer readers to Natarajan et al. (2011). Q.E.D.

Proof of Theorem 1 The proof of Theorem 1 simply follows from Dickinson's(2010) characterization. Condition (i) implies that there exists a set of independent nonnegative vectors $\mathbf{d}^{(j)}, j = 1, ..., m$, that spans $\mathbb{R}^N$ and $\exists \mathbf{d}^{(k)} \in \{\mathbf{d}^{(1)}, \ldots, \mathbf{d}^{(m)}\} > 0$, such that we can find a convex combination of nonnegative rank-one matrices based on $\mathbf{d}^{(j)}$ satisfying

$$\begin{pmatrix} 1 & \boldsymbol{\mu}_d^\intercal \\ \boldsymbol{\mu}_d & \Sigma_d \end{pmatrix} = \sum_j \lambda_j \begin{pmatrix} 1 \\ \mathbf{d}^{(\mathrm{j})} \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{d}^{(\mathrm{j})} \end{pmatrix}^\intercal$$

On the other hand, a convex combination of the set of $\mathbf{x}^{(i)}$ satisfying Condition (ii)

$$\begin{pmatrix} 1 & p^\intercal \\ p & X \end{pmatrix} = \sum_i \eta_i \begin{pmatrix} 1 \\ \mathbf{x}^{(i)} \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{x}^{(i)} \end{pmatrix}^\intercal,$$

is a feasible solution to (9). Then the decomposition

$$\sum_{i,j} \lambda_j \eta_i \begin{pmatrix} 1 \\ \mathbf{d}^{(j)} \\ \mathbf{x}^{(i)} \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{d}^{(j)} \\ \mathbf{x}^{(i)} \end{pmatrix}^\intercal$$

is a natural candidate for membership in the interior of the completely positive cone defined in (9).

Proof of Theorem 2 The main idea of the proof is to construct a strict interior solution in a smaller cone based on the set of $\boldsymbol{\theta}^{(i)}$ in Condition (ii'). Then map the interior point into the relative interior of a completely positive cone in a higher dimension, which is the targeted cone defined in (9).

Our proof exploits the characterization of the relative interior in a cone proposed by Rockafellar (1970).

Theorem 3. *(Rockafellar (1997) Theorem 6.4): A point C_x lies in the relative interior of set $\mathcal{S}$ if and only if for every $C_y \in \mathcal{S}$, $\exists \mu > 1$ such that $(1-\mu)C_y + \mu C_x$ belongs to $\mathcal{S}$.*

Denote the feasible region of (7) as $\mathcal{Y}$

$$\mathcal{Y} = \left\{ \mathbf{x} \,\middle|\, \begin{array}{l} \mathbf{a}_i^\top \mathbf{x} = b_i, \forall i = 1, \ldots, M \\ (\mathbf{h}_i^\top \mathbf{x} + f_i)(\hat{\mathbf{h}}_j^\top \mathbf{x} + \hat{f}_j) = 0, \forall (i,j) \in \mathcal{H} \\ x_i \in \{0,1\}, \forall i \in \mathcal{B} \end{array} \right\}$$

We split $\mathbf{x}$ into two parts: One includes decision variables $\boldsymbol{\theta} \in \mathbb{R}^{n_1}$, and the other includes slack variables $\mathbf{s} \in \mathbb{R}^{N-n_1}$. Then the linear constraint can be rewritten as $\hat{\mathbf{a}}_i^\top \boldsymbol{\theta} + s_i = \hat{b}_i, \forall i \in \mathcal{M}$, where $\mathcal{M} \subset \{1, \ldots, M\}$. Write in matrix form as $\hat{A}\boldsymbol{\theta} + \mathbf{s} = \hat{\mathbf{b}}$. According to Lemma 1 and Lemma 2, the feasible region of (9), denoted as $\mathcal{D}$, can be written in completely positive decomposition form as follows:

$$\mathcal{D} = conv \left\{ \begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix} \begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix}^\top \,\middle|\, \begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix} \in \mathbb{R}_+^{2N+1}, \begin{array}{l} \begin{pmatrix} \frac{\boldsymbol{\gamma}_1}{\alpha} \\ \frac{\boldsymbol{\gamma}_2}{\alpha} \end{pmatrix} \in \mathcal{Y}, \text{for } \alpha > 0 \\ \begin{pmatrix} \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix} = \mathbf{0}, \text{for } \alpha = 0 \end{array} \right\}$$

where $\boldsymbol{\gamma}_1 \in \mathbb{R}_+^{n_1}$, $\boldsymbol{\gamma}_2 \in \mathbb{R}_+^{N-n_1}$, $\boldsymbol{\beta} \in \mathbb{R}_+^N$. Consider the following set

$$\mathcal{D}_2 = conv \left\{ \begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \end{pmatrix} \begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \end{pmatrix}^\top \,\middle|\, \begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \end{pmatrix} \geq \mathbf{0}, \begin{array}{l} \begin{pmatrix} \frac{\boldsymbol{\gamma}_1}{\alpha} \\ \hat{\mathbf{b}} - \hat{A}\frac{\boldsymbol{\gamma}_1}{\alpha} \end{pmatrix} \in \mathcal{Y}, \text{for } \alpha > 0 \\ \boldsymbol{\gamma}_1 = \mathbf{0}, \text{for } \alpha = 0 \end{array} \right\}$$

It is clear to see that $\mathcal{D}$ has a one-to-one mapping to $\mathcal{D}_2$: For each feasible decomposition $\begin{pmatrix} \alpha \\ \mathbf{d} \\ \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix}$ in $\mathcal{D}$, the subvector $\begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \end{pmatrix}$ is also a feasible decomposition in $\mathcal{D}_2$; for each feasible decomposition term $\begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \end{pmatrix}$ in $\mathcal{D}_2$, construct $\boldsymbol{\gamma}_2 = \alpha\hat{\mathbf{b}} - \hat{A}\boldsymbol{\gamma}_1$ such that the corresponding $\begin{pmatrix} \alpha \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix}$ satisfy all the constraints in $\mathcal{D}$.

Condition (i) and Condition (ii') imply that there exists an interior point in $\mathcal{D}_2$, according to Theorem 1. Denote the interior point in $\mathcal{D}_2$ as C_{0x}. In the following, we will show that the corresponding point in $\mathcal{D}$ to C_{0x} lies in the relative interior of $\mathcal{D}$.

The proof is built on the following lemma.

Lemma 3. *Suppose that C_{0x} and C_{0y} are two points in $\mathcal{D}_2$. Their corresponding point in $\mathcal{D}$ is denoted as C_x and C_y, respectively. Consider an affine combination: $C_{0z} := \mu C_{0x} + (1-\mu)C_{0y} \in \mathcal{D}_2$, and denote the corresponding point in $\mathcal{D}$ to point C_{0z} as C_z, then $C_z = \mu C_x + (1-\mu)C_y$.*

Proof: C_{0x} can be written in a decomposed form as follows:

$$C_{0x}=\sum_{k\in\kappa_+}\alpha_k^2\begin{pmatrix}1\\ \frac{\boldsymbol{\beta}_k}{\alpha_k}\\ \frac{\boldsymbol{\gamma}_{1k}}{\alpha_k}\end{pmatrix}\begin{pmatrix}1\\ \frac{\boldsymbol{\beta}_k}{\alpha_k}\\ \frac{\boldsymbol{\gamma}_{1k}}{\alpha_k}\end{pmatrix}^\mathsf{T}+\sum_{k\in\kappa_0}\begin{pmatrix}0\\ \boldsymbol{\beta}_k\\ \mathbf{0}\end{pmatrix}\begin{pmatrix}0\\ \boldsymbol{\beta}_k\\ \mathbf{0}\end{pmatrix}^\mathsf{T}:=\begin{pmatrix}1 & X_{12}^+ & X_{13}^+\\ X_{21}^+ & X_{22}^+ & X_{23}^+\\ X_{31}^+ & X_{32}^+ & X_{33}^+\end{pmatrix}+\begin{pmatrix}0&0&0\\0&X_{22}^0&0\\0&0&0\end{pmatrix}$$

where $\sum_{k\in\kappa_+}\alpha_k^2=1$. The corresponding point in $\mathcal{D}$ can be determined as

$$\begin{aligned}C_x&=\sum_{k\in\kappa_+}\alpha_k^2\begin{pmatrix}1\\ \frac{\boldsymbol{\beta}_k}{\alpha_k}\\ \frac{\boldsymbol{\gamma}_{1k}}{\alpha_k}\\ \hat{\mathbf{b}}-\hat{A}\frac{\boldsymbol{\gamma}_{1k}}{\alpha_k}\end{pmatrix}\begin{pmatrix}1\\ \frac{\boldsymbol{\beta}_k}{\alpha_k}\\ \frac{\boldsymbol{\gamma}_{1k}}{\alpha_k}\\ \hat{\mathbf{b}}-\hat{A}\frac{\boldsymbol{\gamma}_{1k}}{\alpha_k}\end{pmatrix}^\mathsf{T}+\sum_{k\in\kappa_0}\begin{pmatrix}0\\ \boldsymbol{\beta}_k\\ \mathbf{0}\\ \mathbf{0}\end{pmatrix}\begin{pmatrix}0\\ \boldsymbol{\beta}_k\\ \mathbf{0}\\ \mathbf{0}\end{pmatrix}^\mathsf{T}\\ &=\begin{pmatrix}1 & X_{12}^+ & X_{13}^+ & (\hat{\mathbf{b}}-\hat{A}X_{31}^+)^\mathsf{T}\\ X_{21}^+ & X_{22}^++X_{22}^0 & X_{23}^+ & (\hat{\mathbf{b}}X_{12}^+-\hat{A}X_{32}^+)^\mathsf{T}\\ X_{31}^+ & X_{32}^+ & X_{33}^+ & (\hat{\mathbf{b}}X_{13}^+-\hat{A}X_{33}^+)^\mathsf{T}\\ (\hat{\mathbf{b}}-\hat{A}X_{31}^+) & (\hat{\mathbf{b}}X_{12}^+-\hat{A}X_{32}^+) & (\hat{\mathbf{b}}X_{13}^+-\hat{A}X_{33}^+) & (\hat{\mathbf{b}}\hat{\mathbf{b}}^\mathsf{T}-\hat{\mathbf{b}}X_{13}^+\hat{A}^\mathsf{T}-\hat{A}X_{31}^+\hat{\mathbf{b}}^\mathsf{T}+\hat{A}X_{33}^+\hat{A}^\mathsf{T})\end{pmatrix}\end{aligned}$$

Similarly, $C_y(C_z)\in\mathcal{D}$ corresponding to $C_{0y}($ resp. $C_{0z})$ can be written in the same form by replacing X with Y(resp. Z). Therefore, from $C_{0z}=\mu C_{0x}+(1-\mu)C_{0y}$, we can infer that $C_z=\mu C_x+(1-\mu)C_y$. By the generalized Slater constraint qualification, there is no duality gap between (9) and (11). Q.E.D.

Now we are ready to show that the corresponding point in $\mathcal{D}$ to the interior point $C_{0x}\in\mathcal{D}_2$, denoted as C_x, lies in the relative interior in $\mathcal{D}$. Using the characterization proposed by Rockafellar (1970), we consider the term $\mu C_x+(1-\mu)C_y,\mu>1$ for any $C_y\in\mathcal{D}$. Look at the corresponding point in $\mathcal{D}_2$ to $C_y\in\mathcal{D}$, denoted as C_{0y}. Since C_{0x} lies in the interior of $\mathcal{D}_2$, according to the necessary condition of the interior point by Rockafellar (1970), $\exists\mu>1$ such that $\mu C_{0x}+(1-\mu)C_{0y}\in\mathcal{D}_2$. Denote $C_{0z}=\mu C_{0x}+(1-\mu)C_{0y}$. Then according to Lemma 3, the corresponding point to $C_{0z}\in\mathcal{D}_2$ in $\mathcal{D}$, denoted as $C_z\in\mathcal{D}$, satisfies $C_z=\mu C_x+(1-\mu)C_y$. Therefore, $\mu C_x+(1-\mu)C_y\in\mathcal{D},\mu>1$. Then according to the characterization theorem in Rockafellar (1970), C_x lies in the relative interior of $\mathcal{D}$. Q.E.D.

Proof of Proposition 2 Under the network structure and cost structure specified in the condition, the feasible region of problem (7) is specified in (6). Since $\mathbf{y},\mathbf{z}\in\{0,1\}^n$, we can add two sets of valid constraints $\mathbf{y}+\mathbf{s^{(1)}}=\mathbf{1}$ and $\mathbf{z}+\mathbf{s^{(2)}}=\mathbf{1}$, and by replacing $\mathbf{y}$ with $\mathbf{1}-\mathbf{y}$ we can get an equivalent reformulation in (23):

$$\begin{array}{llrcll}Z(\tilde{\mathbf{d}})=\max\limits_{\mathbf{y},\mathbf{z}} & & \sum\limits_{j\in\mathcal{V}_0}\tilde{d}_j(1-y_j) & & & \\ s.t & & y_j+s_j^{(1)} & = & 1, & j\in\mathcal{V}_0\\ & & z_j+s_j^{(2)} & = & 1, & j\in\mathcal{V}_0\\ & & 1-y_j+z_j+s_j & = & c_{sj}, & j\in\mathcal{V}_0\\ & & (1-c_{ij})(1-y_j+z_j)(1-z_i) & = & 0, & (i,j)\in\mathcal{A}_0\\ & & \mathbf{y},\mathbf{z} & \in & \{0,1\}^n & \end{array}\tag{23}$$

It is clear to see that (23) is a specific form of (7) by defining

$$\mathbf{a}_i=\begin{pmatrix}\mathbf{e_i^n}\\ \mathbf{0}_n\\ \mathbf{e_i^n}\\ \mathbf{0}_n\\ \mathbf{0}_n\end{pmatrix}\ \mathbf{a}_{n+i}=\begin{pmatrix}\mathbf{0}_n\\ \mathbf{e_i^n}\\ \mathbf{0}_n\\ \mathbf{e_i^n}\\ \mathbf{0}_n\end{pmatrix}\ \mathbf{a}_{2n+i}=\begin{pmatrix}-\mathbf{e_i^n}\\ \mathbf{e_i^n}\\ \mathbf{0}_n\\ \mathbf{0}_n\\ \mathbf{e_i^n}\end{pmatrix}\ \mathbf{h}_i=\begin{pmatrix}\mathbf{0}_n\\ -\mathbf{e_i^n}\\ \mathbf{0}_n\\ \mathbf{0}_n\\ \mathbf{0}_n\end{pmatrix}\ \hat{\mathbf{h}}_i=\begin{pmatrix}\mathbf{e_i^n}\\ -\mathbf{e_i^n}\\ \mathbf{0}_n\\ \mathbf{0}_n\\ \mathbf{0}_n\end{pmatrix}\ \mathbf{b}=\begin{pmatrix}\mathbf{1}_n\\ \mathbf{1}_n\\ \mathbf{c}_s-\mathbf{1}_n\end{pmatrix}$$

where $\mathbf{e_i^n} \in \mathbb{R}^n$ is a unit vector with ith element equal to 1, $f_i = 1, \hat{f}_i = -1$ for $i = 1, \dots, n$, and $\mathcal{H} = \mathcal{A}_0 \cap \{(i,j) \mid c_{ij} = 0\}$. To see the structure of the copositive matrix in (11) more clearly, we refine dual variable $\boldsymbol{\beta}^{(1)}$, $\boldsymbol{\beta}^{(2)}$ and $\boldsymbol{\beta}^{(3)}$ in (11) as follows:

$$\boldsymbol{\beta}^{(1)} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_3 \end{pmatrix} \; \boldsymbol{\beta}^{(2)} = \begin{pmatrix} \boldsymbol{\beta}_4 \\ \boldsymbol{\beta}_5 \\ \boldsymbol{\beta}_6 \end{pmatrix} \; \boldsymbol{\beta}^{(3)} = \begin{pmatrix} \boldsymbol{\beta}_7 \\ \boldsymbol{\beta}_8 \end{pmatrix}$$

where $\boldsymbol{\beta}_i \in \mathbb{R}^n, i = 1, \dots, 8$. Then

$$\mathbf{w} = \begin{pmatrix} \boldsymbol{\beta}_1 - \boldsymbol{\beta}_3 \\ \boldsymbol{\beta}_2 + \boldsymbol{\beta}_3 \\ \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_3 \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}_7 \\ \boldsymbol{\beta}_8 \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{pmatrix} + \begin{pmatrix} (H \circ \Gamma)^\mathsf{T} \mathbf{1}_n \\ (H \circ \Gamma)\mathbf{1}_n - (H \circ \Gamma)^\mathsf{T} \mathbf{1}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{pmatrix}$$

$$W = \begin{pmatrix} diag(\boldsymbol{\beta}_4 + \boldsymbol{\beta}_6 + \boldsymbol{\beta}_7) & -\frac{1}{2}(H \circ \Gamma)^\mathsf{T} - diag(\boldsymbol{\beta}_6) & diag(\boldsymbol{\beta}_4) & O & -diag(\boldsymbol{\beta}_6) \\ -\frac{1}{2}(H \circ \Gamma) - diag(\boldsymbol{\beta}_6) & diag(\boldsymbol{\beta}_5 + \boldsymbol{\beta}_6 + \boldsymbol{\beta}_8) + \frac{1}{2}(H \circ \Gamma + (H \circ \Gamma)^\mathsf{T}) & O & diag(\boldsymbol{\beta}_5) & diag(\boldsymbol{\beta}_6) \\ diag(\boldsymbol{\beta}_4) & O & diag(\boldsymbol{\beta}_4) & O & O \\ O & diag(\boldsymbol{\beta}_5) & O & diag(\boldsymbol{\beta}_5) & O \\ -diag(\boldsymbol{\beta}_6) & diag(\boldsymbol{\beta}_6) & O & O & diag(\boldsymbol{\beta}_6) \end{pmatrix}$$

The objective in (11) becomes

$$\alpha_0 + \boldsymbol{\mu}^\mathsf{T} \boldsymbol{\beta}_0 + \Sigma \bullet \Gamma_0 + \mathbf{1}_n^\mathsf{T}(\boldsymbol{\beta}_1 + \boldsymbol{\beta}_2 + \boldsymbol{\beta}_4 + \boldsymbol{\beta}_5) + (\mathbf{c}_s - \mathbf{1}_n)^\mathsf{T} \boldsymbol{\beta}_3 + (\mathbf{c}_s - \mathbf{1}_n)^\mathsf{T} diag(\boldsymbol{\beta}_6)(\mathbf{c}_s - \mathbf{1}_n) + \mathbf{1}_n^\mathsf{T}(H \circ \Gamma)\mathbf{1}_n$$

where H is an indicator matrix, $H_{i,j} = 1$ if $(i,j) \in \mathcal{H}$.

The key element of this proof is to construct a feasible solution to (11) after deleting arc (a,b)—i.e., adding constraint $\Gamma_{ab} = 0$—whose objective is increased by not more than $(\frac{1}{4} + c_{sb})|\Gamma^*_{ab}|$ compared to the optimal value of (11) before deleting arc (a,b). Denote the optimal copositive matrix in (11) before deleting arc (a,b) as $\mathcal{C}^*$. Then according to the definition of the copositive matrix, $\mathbf{v}^\mathsf{T} \mathcal{C}^* \mathbf{v} \geq 0, \forall \mathbf{v} \geq \mathbf{0}$. Refine $\mathbf{v}$ to be

$$\mathbf{v} = \begin{pmatrix} 1 & \mathbf{v}_\mu^\mathsf{T} & \mathbf{v}_y^\mathsf{T} & \mathbf{v}_z^\mathsf{T} & \mathbf{v}_1^\mathsf{T} & \mathbf{v}_2^\mathsf{T} & \mathbf{v}_3^\mathsf{T} \end{pmatrix}^\mathsf{T}$$

where $\mathbf{v}_\mu, \mathbf{v}_y, \mathbf{v}_z, \mathbf{v}_i \in \mathbb{R}^n_+, i = 1,2,3$. List the terms in $\mathbf{v}^\mathsf{T} \mathcal{C}^* \mathbf{v}$ related to $\Gamma^*_{a,b}$ as follows:

$$\begin{aligned} (\beta^*_{1b} - \beta^*_{3b} - \beta^*_{7b} + \Gamma^*_{a,b}) v_{yb} + (\beta^*_{2a} + \beta^*_{3a} - \beta^*_{8a} + \Gamma^*_{a,b}) v_{za} + (\beta^*_{2b} + \beta^*_{3b} - \beta^*_{8b} - \Gamma^*_{a,b}) v_{zb} \\ -\Gamma^*_{a,b} v_{za} v_{yb} + \Gamma^*_{a,b} v_{za} v_{zb} + \beta^*_{8a} v_{za}^2 + \beta^*_{8b} v_{zb}^2 + \beta^*_{7a} v_{ya}^2 + \beta^*_{7b} v_{yb}^2 \\ + \beta^*_{1b} v_{1b} + \beta^*_{2a} v_{2a} + \beta^*_{2b} v_{2b} + \beta^*_{3a} v_{3a} + \beta^*_{3b} v_{3b} \end{aligned}$$

Denote the optimal value of Problem (11) under graph $\mathcal{G}$ as $Z_{CD}(\mathcal{G})$ and the optimal value of Problem (11) under graph $\mathcal{G} \setminus (a,b)$ as $Z^r_{CD}(\mathcal{G})$. We separately consider the two cases: $\Gamma^*_{ab} \leq 0$ and $\Gamma^*_{ab} > 0$.

(1) $\Gamma^*_{ab} \leq 0$

Consider such a set of values of dual variables, $\hat{\alpha}_0 = \alpha^*_0$, $\hat{\boldsymbol{\beta}}_i = \boldsymbol{\beta}^*_i, i = 0, \dots, 8$, $\hat{\Gamma} = \Gamma^*$,$\hat{\Gamma}_0 = \Gamma^*_0$ except

$$\begin{aligned} \hat{\beta}_{3b} = \beta^*_{3b} - \Gamma^*_{ab}, \quad \hat{\beta}_{8a} = \beta^*_{8a} - \Gamma^*_{ab} \\ \hat{\beta}_{7b} = \beta^*_{7b} - \frac{1}{4}\Gamma^*_{ab}, \; \hat{\beta}_{1b} = \beta^*_{1b} - \tfrac{1}{4}\Gamma^*_{ab}, \; \hat{\Gamma}_{ab} = 0 \end{aligned}$$

Denote the copositive matrix formed by $\hat{\alpha}_0$, $\hat{\boldsymbol{\beta}}_i, i = 0, \dots, 8$, $\hat{\Gamma}$,$\hat{\Gamma}_0$ as $\hat{\mathcal{C}}$, then

$$\mathbf{v}^\mathsf{T} \mathcal{C}^* \mathbf{v} - \mathbf{v}^\mathsf{T} \hat{\mathcal{C}} \mathbf{v} = \Gamma^*_{ab}(\frac{1}{4} v_{yb}^2 + v_{za}^2 - v_{za} v_{yb} + v_{za} v_{zb} + v_{3b} + \frac{1}{4} v_{1b}) \leq 0$$

The inequality holds due to $\Gamma^*_{ab} \le 0$, $\mathbf{v} \ge \mathbf{0}$ and $\frac{1}{4}v_{yb}^2 + v_{za}^2 - v_{za}v_{yb} = (\frac{1}{2}v_{yb} - v_{za})^2 \ge 0$. Hence $\hat{\mathcal{C}}$ is a feasible solution to (11). And the objective change is

$$\Delta = -\Gamma^*_{ab} + (\hat{\beta}_{1b} - \beta^*_{1b}) + (c_{sb} - 1)(\hat{\beta}_{3b} - \beta^*_{3b}) = -(\frac{1}{4} + c_{sb})\Gamma^*_{ab} = (\frac{1}{4} + c_{sb})|\Gamma^*_{ab}|$$

(2) $\Gamma^*_{ab} > 0$

Consider such a set of values of dual variables, $\hat{\alpha}_0 = \alpha^*_0$, $\hat{\boldsymbol{\beta}}_i = \boldsymbol{\beta}^*_i, i = 0, \ldots, 8$, $\hat{\Gamma} = \Gamma^*$,$\hat{\Gamma}_0 = \Gamma^*_0$ except

$$\hat{\beta}_{1b} = \beta^*_{1b} + \Gamma^*_{ab},\ \hat{\beta}_{8a} = \beta^*_{8a} + \Gamma^*_{ab}$$
$$\hat{\beta}_{3a} = \beta^*_{3a} + 2\Gamma^*_{ab},\ \hat{\beta}_{8b} = \beta^*_{8b} + \Gamma^*_{ab},\ \hat{\Gamma}_{ab} = 0.$$

Then

$$\mathbf{v}^\mathsf{T}\mathcal{C}^*\mathbf{v} - \mathbf{v}^\mathsf{T}\hat{\mathcal{C}}\mathbf{v} = \Gamma^*_{ab}(-v_{za}^2 - v_{zb}^2 + v_{za}v_{zb} - v_{za}v_{yb} - 2v_{3a} - v_{1b}) \le 0$$

The inequality holds due to $\Gamma^*_{ab} > 0$, $\mathbf{v} \ge \mathbf{0}$ and $-v_{za}^2 - v_{zb}^2 + v_{za}v_{zb} = -(v_{zb} - \frac{1}{2}v_{za})^2 - \frac{3}{4}v_{za}^2 \le 0$. And the objective change is

$$\Delta = -\Gamma^*_{ab} + (\hat{\beta}_{1b} - \beta^*_{1b}) + (c_{sb} - 1)(\hat{\beta}_{3a} - \beta^*_{3a}) = 2(c_{sb} - 1)\Gamma^*_{ab} \le 0$$

since $c_{sb} \in \{0,1\}$. Therefore, in both cases, we manage to construct a feasible solution to (11) such that the corresponding objective change $\Delta \le (\frac{1}{4} + c_{sb})|\Gamma^*_{ab}|$. Notice that (11) is a minimization problem, hence $Z^r_{CD}(\mathcal{G}) - Z_{CD}(\mathcal{G}) \le \Delta \le (\frac{1}{4} + c_{sb})|\Gamma^*_{ab}|$. Q.E.D.

Proof of Proposition 3 Condition (ii) fails in the roving team deployment problem due to the presence of a slack variable. Hence to prove strong duality, we need to check whether the conditions in Theorem 2 hold. We write the feasible region in (7) in the roving team deployment problem as follows:

$$\mathcal{Y} = \left\{ \begin{pmatrix} \mathbf{y} \\ \mathbf{z} \\ \mathbf{s^{(1)}} \\ \mathbf{s^{(2)}} \\ \mathbf{s} \end{pmatrix} \middle| \begin{array}{rll} y_j + s_j^{(1)} &= 1, & j \in \mathcal{V}_0 \\ z_j + s_j^{(2)} &= 1, & j \in \mathcal{V}_0 \\ -y_j + z_j + s_j &= 0, & j \in \mathcal{V}_0 \\ (1 - y_j + z_j)(1 - z_i) &= 0, & (i,j) \in \mathcal{A}_0 \\ \mathbf{y}, \mathbf{z} &\in \{0,1\}^n & \end{array} \right\}$$

Ignoring slack variables, the feasible solution $(\mathbf{y}, \mathbf{z})$ satisfies:

$$z_j \le y_j, \quad j \in \mathcal{V}_0 \tag{24}$$

$$(1 - y_j + z_j)(1 - z_i) = 0, \quad (i,j) \in \mathcal{A}_0 \tag{25}$$

$$\mathbf{y}, \mathbf{z} \in \{0,1\}^{2n} \tag{26}$$

For each node i, we define the predecessor of node i as follows: Node j is the predecessor of node i if there exists a path from j to i. Similarly, node j is defined as the successor of node i if there exists a path from i to j. Denote $\mathbf{e}^{(\mathcal{S}_i)}$ as the indicator vector for the successor of node i, i.e., $\mathbf{e}_j^{(\mathcal{S}_i)} = 1$ if node j is the successor of node i; otherwise, $\mathbf{e}_j^{(\mathcal{S}_i)} = 0$. And we define $\mathbf{e}_i^{(\mathcal{S}_i)} = 1$.

Consider such a set of vectors: $\mathbf{v}_0 = \begin{pmatrix} \mathbf{1}_n \\ \mathbf{1}_n \end{pmatrix}$,$\mathbf{v}_i = \begin{pmatrix} \mathbf{1}_n \\ \mathbf{1}_n \end{pmatrix} - \begin{pmatrix} \mathbf{e^n}_i \\ \mathbf{e}^{(\mathcal{S}_i)} \end{pmatrix}, \forall i = 1, .., n$,$\mathbf{v}_{n+i} = \begin{pmatrix} \mathbf{1}_n \\ \mathbf{1}_n \end{pmatrix} - \begin{pmatrix} \mathbf{0}_n \\ \mathbf{e}^{(\mathcal{S}_i)} \end{pmatrix}, \forall i = 1, .., n.$

where the first subvector in $\mathbb{R}^n$ refers to the y value and the second subvector refers to the z value. We first show that they satisfy Constraints (24) to (26). Notice that for each fixed node $i, i=1,...,n$, the rest of the nodes can be divided into three groups: predecessors (denoted as $\mathcal{P}(i)$), successors (denoted as $\mathcal{S}(i)$), and other "irrelevant" nodes.

Consider each fixed vector $\mathbf{v}_i, i=1,...,n$. Where $z_i=0$, y_i can be either 1 or 0. For each pairwise nodes $(a,b)\in\mathcal{A}_0$, if $a\in\mathcal{S}(i)$, then $z_a=0, y_a=1$ and $z_b=0, y_b=1$, satisfying all the constraints from (24) to (26). If $a\in\mathcal{P}(i)$, then $z_a=1, y_a=1$, then y_b, z_b can be any $0-1$ value as long as $y_b\geq z_b$, hence in the constructed solution, constraints (24) to (26) hold. If $a\in\{i\}\cup\mathcal{P}(i)\cup\mathcal{S}(i), b\notin\{i\}\cup\mathcal{P}(i)\cup\mathcal{S}(i)$, then node a must be the predecessor to i, i.e., $a\in\mathcal{P}(i)$, then $y_a=1, z_a=1$, hence $y_b=1, z_b=1$ also satisfy all the constraints from (24) to (26). If $a\notin\{i\}\cup\mathcal{P}(i)\cup\mathcal{S}(i), b\in\{i\}\cup\mathcal{P}(i)\cup\mathcal{S}(i)$, then node b must be the successor to i, i.e., $b\in\mathcal{S}(i)$, then $y_b=1, z_b=0$, while $y_a=1, z_a=1$ still satisfy all the constraints. In the last case $a,b\notin\{i\}\cup\mathcal{P}(i)\cup\mathcal{S}(i)$, then $y_a=1, z_a=1$, $y_b=1, z_b=1$ is still feasible. Using a similar argument, we can show that $\mathbf{v}_{n+i}, i=1,...,n$ are also feasible to $\mathcal{D}_2$.

Then we show that these $2n+1$ vectors $\mathbf{v}_0$, $\mathbf{v}_i$ and $\mathbf{v}_{n+i}, i=1,...,n$ span $\mathbb{R}^{2n}$. Denote J as the matrix in $\mathbb{R}^{n\times n}$ with all the entries equal to be 1. Define the Reachability matrix R as follows: $R_{ij}=1$ if there exists a path from i to j and we let $R_{ii}=1$. Then we can write $\mathbf{v}_i, \mathbf{v}_{n+i}, i=1,...,n$ in matrix form as

$$\begin{pmatrix} J-I & J \\ J-R^{\mathsf{T}} & J-R^{\mathsf{T}} \end{pmatrix}$$

Before we move on to the proof of independence, we first show

LEMMA 4. *Denote I as the identity matrix in $\mathbb{R}^{n\times n}$. If $B_1\in\mathbb{R}^{n\times n}, B_2\in\mathbb{R}^{n\times n}$ are both nonsingular, then $\begin{pmatrix} B_1 & B_3 \\ B_4 & B_2 \end{pmatrix}$ has the same rank as $\begin{pmatrix} I & A_1 \\ 0 & -A_2A_1+I \end{pmatrix}$, where A_1, A_2 satisfy $A_1=B_1^{-1}B_3$, $A_2=B_2^{-1}B_4$.*

Proof: Since B_1 is nonsingular, the columns $\mathbf{v}_i, i=1,...,n$ are linearly independent. And they can be regarded as a set of basis in $\mathbb{R}^{n\times n}$, hence each column in B_3 can be represented as a linear combination of $\mathbf{v}_i, i=1,...,n$, e.g, the first column $B_{31}=a_{11}^{(1)}\mathbf{v}_1+a_{21}^{(1)}\mathbf{v}_2+...+a_{n1}^{(1)}\mathbf{v}_n=B_1\begin{pmatrix} a_{11}^{(1)} \\ a_{21}^{(1)} \\ ... \\ a_{n1}^{(1)} \end{pmatrix}$. Denote the coefficient matrix as A_1, then $B_3=B_1A_1$; similarly, there exists a unique coeffcieint matrix A_2 satisfying $B_4=B_2A_2$. Consider a linear equation system $\begin{pmatrix} B_1 & B_1A_1 \\ B_2A_2 & B_2 \end{pmatrix}\begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{pmatrix}=0$, where $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}\in\mathbb{R}^n$. We write the linear equation system in the form of a linear combination of vectors

$$x_1^{(1)}\mathbf{v}_1+...+x_n^{(1)}\mathbf{v}_1+(a_{11}^{(1)}\mathbf{v}_1+...+a_{n1}^{(1)}\mathbf{v}_n)x_1^{(2)}+...+(a_{1n}^{(1)}\mathbf{v}_1+...+a_{nn}^{(1)}\mathbf{v}_n)x_n^{(2)}=0$$

Regroup it to get

$$(x_1^{(1)}+a_{11}^{(1)}x_1^{(2)}+...+a_{1n}^{(1)}x_n^{(2)})\mathbf{v}_1+...+(x_n^{(1)}+a_{n1}^{(1)}x_1^{(2)}+...+a_{nn}^{(1)}x_n^{(2)})\mathbf{v}_n=0$$

Since $\mathbf{v}_i, i=1,...,n$ are independent, all the solutions of the linear system should satisfy $x_i^{(1)}+a_{i1}^{(1)}x_1^{(2)}+...+a_{in}^{(1)}x_n^{(2)}=0, i=1,...,n$ Write this in matrix form as $\begin{pmatrix} I & A_1 \end{pmatrix}\begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{pmatrix}=0$. Applying a similar procedure, we can infer $\begin{pmatrix} A_2 & I \end{pmatrix}\begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{pmatrix}=0$. Therefore, the linear equation system is equivalent to $\begin{pmatrix} I & A_1 \\ A_2 & I \end{pmatrix}\begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{pmatrix}=0$. Hence$\begin{pmatrix} B_1 & B_3 \\ B_4 & B_2 \end{pmatrix}$ has the same rank with $\begin{pmatrix} I & A_1 \\ A_2 & I \end{pmatrix}$.

Note that $\begin{pmatrix} I & A_1 \\ 0 & -A_2A_1+I \end{pmatrix}$ can be obtained by applying a sequence of elementary row operations to the matrix $\begin{pmatrix} I & A_1 \\ A_2 & I \end{pmatrix}$. Since applying elementary row operations will not change the rank of the matrix $\begin{pmatrix} B_1 & B_3 \\ B_4 & B_2 \end{pmatrix}$ has the same row rank as $\begin{pmatrix} I & A_1 \\ 0 & -A_2A_1+I \end{pmatrix}$ Q.E.D.

Note that $J-I$ is nonsingular and we assume that $J-R^{\mathsf{T}}$ is also nonsingular. According to the lemma above, the matrix $\begin{pmatrix} J-I & J \\ J-R^{\mathsf{T}} & J-R^{\mathsf{T}} \end{pmatrix}$ has the same rank with $\begin{pmatrix} I & (J-I)^{-1}J \\ 0 & I-(J-I)^{-1}J \end{pmatrix}$, which is nonsingular. Hence $\begin{pmatrix} J-I & J \\ J-R^{\mathsf{T}} & J-R^{\mathsf{T}} \end{pmatrix}$ is of full row rank. Therefore the constructed vectors $\mathbf{v}_{j1}, \mathbf{v}_{j2}, j=1,...,n$ and $\mathbf{v}_0$ span $\mathbb{R}^{2n}$, with $\mathbf{v}_0$ strictly positive.

Therefore, the constructed $\mathbf{v}_i, i=0,1,\ldots,2n$ satisfy Condition (ii'). On the other hand, Condition (1) implies the moment matrix lies in the interior of a completely positive cone (satisfying Dickinson (2010)'s characterization (12) and moment conditions). Therefore, strong duality holds according to Theorem 2. Q.E.D.

Proof of Proposition 4 Condition (ii) fails in the process flexibility problem due to the presence of slack variables. Hence, to prove strong duality, we need to check whether the conditions in Theorem 2 hold. We write the feasible region in (7) in the process felxibility problem as follows:

$$\mathcal{Y} = \left\{ \begin{pmatrix} \mathbf{y} \\ \mathbf{z} \\ \mathbf{s^{(1)}} \\ \mathbf{s^{(2)}} \end{pmatrix} \middle| \begin{array}{rll} y_j + s_j^{(1)} = 1, & & j \in \mathcal{J} \\ z_j + s_j^{(2)} = 1, & & j \in \mathcal{I} \\ (1-y_j)(1-z_i) = 0, & & (i,j) \in \mathcal{A}_0 \\ \mathbf{y} \in \{0,1\}^m, & \mathbf{z}\{0,1\}^n & \end{array} \right\}$$

Ignoring slack variables, the feasible solution $(\mathbf{y},\mathbf{z})$ satisfies:

$$(1-y_j)(1-z_i) = 0, \quad (i,j) \in \mathcal{A}_0 \tag{27}$$

$$\mathbf{y} \in \{0,1\}^m, \ \ \mathbf{z}\{0,1\}^n \tag{28}$$

Consider such a set of vectors: $\mathbf{v}_1 = \begin{pmatrix} \mathbf{1}_m \\ \mathbf{1}_n \end{pmatrix}, \mathbf{v}_{j+1} = \begin{pmatrix} \mathbf{1}_m \\ \mathbf{1}_n \end{pmatrix} - \mathbf{e}_j, \forall j = 1,..,m-1, \mathbf{v}_{m+i} = \begin{pmatrix} \mathbf{1}_m \\ \mathbf{1}_n \end{pmatrix} - \mathbf{e}_{m+i}, \forall i = 1,..,n$, where the first subvector in $\mathbb{R}^m$ refers to y value and the second subvector n $\mathbb{R}^n$ refers to z value. First we show that this set of vectors satisfies Constraints (27) and (28). We let each element in $\mathbf{v}_l$ represent a node in $\mathcal{V}_0$: $v_{l,m+i}$ represents supply node i, and $v_{l,j}$ corresponds to demand node j. Then for any $(i,j) \in \mathcal{A}_0$, at least one of $\mathbf{v}_{l,m+i}$ and $\mathbf{v}_{l,j}$ is 1. In other words, the constructed vectors satisfy Constraint (27). Besides that, all the vectors $\mathbf{v}_l$ are nonnegative and binary, which satisfies Constraint (28). Notice that $\mathbf{v}_l, \forall l = 1,..,m+n$ are linearly independent, and the first vector $\mathbf{v}_1$ is strictly positive. Therefore, the constructed $\mathbf{v}_i, i=1,\ldots,m+n$ satisfy Condition (ii'). Combined with the condition that the moment lies in the interior of completely positive cone $\mathcal{C}_m^*$, strong duality holds according to Theorem 2. Q.E.D.

Proof of Proposition 5 It is clear to see that (19) is a specific form of (7) by defining

$$\mathbf{a}_i = \begin{pmatrix} \mathbf{e^m}_i \\ \mathbf{0}_n \\ \mathbf{e^m}_i \\ \mathbf{0}_n \end{pmatrix} \ \mathbf{a}_{m+i} = \begin{pmatrix} \mathbf{0}_m \\ \mathbf{e^n}_i \\ \mathbf{0}_m \\ \mathbf{e^n}_i \end{pmatrix} \ \mathbf{h}_i = \begin{pmatrix} \mathbf{0}_m \\ -\mathbf{e^n}_i \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix} \ \hat{\mathbf{h}}_i = \begin{pmatrix} \mathbf{e^m}_i \\ \mathbf{0}_n \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix} \ \mathbf{b} = \begin{pmatrix} \mathbf{1}_m \\ \mathbf{1}_n \end{pmatrix}$$

$f_i = 1$, for $i = 1,\ldots,n$ and $\hat{f}_i = -1$ for $i = 1,\ldots,m$. And $\mathcal{H} = \mathcal{A}_0$. Where $\mathbf{e^n}_i(\mathbf{e^m}_i)$ denote unit vector in $\mathbb{R}^n(\mathbb{R}^m)$. To see the structure of copositive matrix in (11) more clearly, we refine dual variable $\boldsymbol{\beta}^{(1)}$, $\boldsymbol{\beta}^{(2)}$ and $\boldsymbol{\beta}^{(3)}$ in (11) as follows:

$$\boldsymbol{\beta}^{(1)} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \end{pmatrix} \ \boldsymbol{\beta}^{(2)} = \begin{pmatrix} \boldsymbol{\beta}_3 \\ \boldsymbol{\beta}_4 \end{pmatrix} \ \boldsymbol{\beta}^{(3)} = \begin{pmatrix} \boldsymbol{\beta}_5 \\ \boldsymbol{\beta}_6 \end{pmatrix}$$

where $\boldsymbol{\beta}_1, \boldsymbol{\beta}_3, \boldsymbol{\beta}_5 \in \mathbb{R}^m$ and $\boldsymbol{\beta}_2, \boldsymbol{\beta}_4, \boldsymbol{\beta}_6 \in \mathbb{R}^n$. Then

$$\mathbf{w} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}_5 \\ \boldsymbol{\beta}_6 \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix} + \begin{pmatrix} (H\circ\Gamma)^\mathsf{T}\mathbf{1}_n \\ (H\circ\Gamma)\mathbf{1}_m \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix}$$

$$W = \begin{pmatrix} diag(\boldsymbol{\beta}_3+\boldsymbol{\beta}_5) & -\frac{1}{2}(H\circ\Gamma)^\mathsf{T} & diag(\boldsymbol{\beta}_3) & O \\ -\frac{1}{2}(H\circ\Gamma) & diag(\boldsymbol{\beta}_4+\boldsymbol{\beta}_6) & O & diag(\boldsymbol{\beta}_4) \\ diag(\boldsymbol{\beta}_3) & O & diag(\boldsymbol{\beta}_3) & O \\ O & diag(\boldsymbol{\beta}_4) & O & diag(\boldsymbol{\beta}_4) \end{pmatrix}$$

And objective in (11) becomes

$$\alpha_0 + \boldsymbol{\mu}_d^\mathsf{T}\boldsymbol{\beta}_0 + \Sigma_d \bullet \Gamma_0 + \mathbf{1}_m^\mathsf{T}(\boldsymbol{\beta}_1+\boldsymbol{\beta}_3) + \mathbf{1}_n^\mathsf{T}(\boldsymbol{\beta}_2+\boldsymbol{\beta}_4) + \mathbf{1}_n^\mathsf{T}(H\circ\Gamma)\mathbf{1}_m$$

where H is an indicator matrix, $H_{i,j}=1$ if $(i,j)\in\mathcal{H}$.

The key element in this proof is to construct a feasible solution to (11) after deleting arc (a,b), with the objective to increase not more than $|\Gamma^*_{ab}|$ compared to the optimal value of (11) before deleting arc (a,b). Denote the optimal copositive matrix in (11) before deleting arc (a,b) as $\mathcal{C}^*$. Then according to the definition of the copositive matrix, $\mathbf{v}^\mathsf{T}\mathcal{C}^*\mathbf{v}\geq 0, \forall \mathbf{v}\geq\mathbf{0}$. Refine $\mathbf{v}$ to be

$$\mathbf{v} = \begin{pmatrix} 1 & \mathbf{v}_\mu^\mathsf{T} & \mathbf{v}_y^\mathsf{T} & \mathbf{v}_z^\mathsf{T} & \mathbf{v}_1^\mathsf{T} & \mathbf{v}_2^\mathsf{T} \end{pmatrix}^\mathsf{T}$$

where $\mathbf{v}_\mu, \mathbf{v}_y, \mathbf{v}_z, \mathbf{v}_i \in \mathbb{R}^n_+, i=1,2$. And list the terms in $\mathbf{v}^\mathsf{T}\mathcal{C}^*\mathbf{v}$ related to $\Gamma^*_{a,b}$ as follows:

$$(\beta^*_{1b}-\beta^*_{5b}+\Gamma^*_{a,b})v_{yb} + (\beta^*_{2a}-\beta^*_{6a}+\Gamma^*_{a,b})v_{za} - \Gamma^*_{a,b}v_{za}v_{yb} + \beta^*_{6a}v^2_{za} + \beta^*_{5b}v^2_{5b} + \beta^*_{1b}v_{1b} + \beta^*_{2a}v_{2a}$$

Denote the optimal value of Problem (11) under graph $\mathcal{G}$ as $Z_{CD}(\mathcal{G})$ and the optimal value of Problem (11) under graph $\mathcal{G}\setminus(a,b)$ as $Z^r_{CD}(\mathcal{G})$. We separately consider the two cases: $\Gamma^*_{ab}\leq 0$ and $\Gamma^*_{ab}>0$.

(1) $\Gamma^*_{ab}\leq 0$

Consider such a set of values of dual variables, $\hat\alpha_0=\alpha^*_0$, $\hat{\boldsymbol{\beta}}_i=\boldsymbol{\beta}^*_i, i=0,\ldots,6$, $\hat\Gamma=\Gamma^*$,$\hat\Gamma_0=\Gamma^*_0$ except

$$\hat\beta_{5b}=\beta^*_{5b}-\Gamma^*_{ab},\ \hat\beta_{6a}=\beta^*_{6a}-\Gamma^*_{ab},\ \hat\Gamma_{ab}=0$$

Denote the copositive matrix formed by $\hat\alpha_0$, $\hat{\boldsymbol{\beta}}_i, i=0,\ldots,8$, $\hat\Gamma$,$\hat\Gamma_0$ as $\hat{\mathcal{C}}$, then

$$\mathbf{v}^\mathsf{T}\mathcal{C}^*\mathbf{v} - \mathbf{v}^\mathsf{T}\hat{\mathcal{C}}\mathbf{v} = \Gamma^*_{ab}(v^2_{yb}+v^2_{za}-v_{za}v_{yb})\leq 0$$

The inequality holds due to $\Gamma^*_{ab}\leq 0$, $v^2_{yb}+v^2_{za}-v_{za}v_{yb}=\frac{3}{4}v^2_{yb}+(\frac{1}{2}v_{yb}-v_{za})^2\geq 0$. Hence $\hat{\mathcal{C}}$ is a feasible solution to (11). And the objective change is

$$\Delta = -\Gamma^*_{ab} = |\Gamma^*_{ab}|$$

(2) $\Gamma^*_{ab}>0$

Consider such a set of values of dual variables, $\hat\alpha_0=\alpha^*_0$, $\hat{\boldsymbol{\beta}}_i=\boldsymbol{\beta}^*_i, i=0,\ldots,6$, $\hat\Gamma=\Gamma^*$,$\hat\Gamma_0=\Gamma^*_0$ except

$$\hat\beta_{1b}=\beta^*_{1b}+\Gamma^*_{ab},\ \hat\beta_{2a}=\beta^*_{2a}+\Gamma^*_{ab},\ \hat\Gamma_{ab}=0$$

then

$$\mathbf{v}^\mathsf{T}\mathcal{C}^*\mathbf{v} - \mathbf{v}^\mathsf{T}\hat{\mathcal{C}}\mathbf{v} = \Gamma^*_{ab}(-v_{za}v_{yb}-v_{2a}-v_{1b})\leq 0$$

The inequality holds due to $\Gamma^*_{ab}>0$, $\mathbf{v}\geq\mathbf{0}$. And the objective change is

$$\Delta = -\Gamma^*_{ab} + (\hat\beta_{1b}-\beta^*_{1b}) + (\hat\beta_{2a}-\beta^*_{2a}) = \Gamma^*_{ab}$$

Therefore, in both cases, we manage to construct a feasible solution to (11) such that the corresponding objective change $\Delta=|\Gamma^*_{ab}|$. Notice that (11) is a minimization problem, hence $Z^r_{CD}(\mathcal{G})-Z_{CD}(\mathcal{G})\leq\Delta=|\Gamma^*_{ab}|$. Q.E.D.

Appendix B: Detailed Formulation

(1) Conic Formulation in Roving Team Deployment Problem

Define

$$\mathbf{a}_i = \begin{pmatrix} \mathbf{e^n}_i \\ \mathbf{0}_n \\ \mathbf{e^n}_i \\ \mathbf{0}_n \\ \mathbf{0}_n \end{pmatrix} \ \mathbf{a}_{n+i} = \begin{pmatrix} \mathbf{0}_n \\ \mathbf{e^n}_i \\ \mathbf{0}_n \\ \mathbf{e^n}_i \\ \mathbf{0}_n \end{pmatrix} \ \mathbf{a}_{2n+i} = \begin{pmatrix} -\mathbf{e^n}_i \\ \mathbf{e^n}_i \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{e_i} \end{pmatrix}$$

where $\mathbf{e^n}_i \in \mathbb{R}^n$ is a unit vector. The completely positive program equivalent to the worst-case expected value of (14) can be written as

$$\begin{array}{llcl} Z_{C1} = \sup & \mathbf{1}^\top \boldsymbol{\mu}_d - I \bullet Y & & \\ s.t. & \mathbf{a}_i^\top \mathbf{p} & = & 1, \forall i \in \{1,..,3n\} \\ & \mathbf{a}_i^\top X \mathbf{a}_i & = & 1, \forall i \in \{1,..,3n\} \\ & X_{ii} & = & p_i, \forall i \in \{1,..,2n\} \\ & p_j - p_{n+j} + p_{n+i} - X_{n+i,j} + X_{n+i,n+j} & = & 1, \forall i,j \in \mathcal{A}_0 \\ & (\boldsymbol{\mu}, \Sigma) & \in & \Omega(\boldsymbol{\mu}_d, \Sigma_d) \\ & \begin{pmatrix} 1 & \boldsymbol{\mu}^\top & \mathbf{p}^\top \\ \boldsymbol{\mu} & \Sigma & Y^\top \\ \mathbf{p} & Y & X \end{pmatrix} & \succcurlyeq_{cp} & 0 \end{array} \tag{29}$$

Refine dual variable $\boldsymbol{\beta}^{(1)}$, $\boldsymbol{\beta}^{(2)}$ and $\boldsymbol{\beta}^{(3)}$ in (11) as follows:

$$\boldsymbol{\beta}^{(1)} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_3 \end{pmatrix} \ \boldsymbol{\beta}^{(2)} = \begin{pmatrix} \boldsymbol{\beta}_4 \\ \boldsymbol{\beta}_5 \\ \boldsymbol{\beta}_6 \end{pmatrix} \ \boldsymbol{\beta}^{(3)} = \begin{pmatrix} \boldsymbol{\beta}_7 \\ \boldsymbol{\beta}_8 \end{pmatrix}$$

where $\boldsymbol{\beta}_i \in \mathbb{R}^n, i = 1, \ldots, 8$. Denote the dual variable of each constraint in (10) as $\boldsymbol{\beta}_0 \in \mathbb{R}^n$, $\Sigma_0 \in \mathbb{R}^{n\times n}$, $\boldsymbol{\eta}^{(i)} \in \mathbb{R}^n, i = 1, \ldots, 4$ Define

$$\mathbf{w} = \begin{pmatrix} \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{pmatrix} + \begin{pmatrix} \boldsymbol{\beta}_1 - \boldsymbol{\beta}_3 \\ \boldsymbol{\beta}_2 + \boldsymbol{\beta}_3 \\ \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_3 \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}_7 \\ \boldsymbol{\beta}_8 \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{pmatrix} + \begin{pmatrix} (H \circ \Gamma)^\top \mathbf{1}_n \\ (H \circ \Gamma)\mathbf{1}_n - (H \circ \Gamma)^\top \mathbf{1}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{pmatrix}$$

$$W = \begin{pmatrix} diag(\boldsymbol{\beta}_4 + \boldsymbol{\beta}_6 + \boldsymbol{\beta}_7) & -\frac{1}{2}(H \circ \Gamma)^\top - diag(\boldsymbol{\beta}_6) & diag(\boldsymbol{\beta}_4) & O & -diag(\boldsymbol{\beta}_6) \\ -\frac{1}{2}(H \circ \Gamma)^\top - diag(\boldsymbol{\beta}_6) & diag(\boldsymbol{\beta}_5 + \boldsymbol{\beta}_6 + \boldsymbol{\beta}_8) + \frac{1}{2}(H \circ \Gamma + (H \circ \Gamma)^\top) & O & diag(\boldsymbol{\beta}_5) & diag(\boldsymbol{\beta}_6) \\ diag(\boldsymbol{\beta}_4) & O & diag(\boldsymbol{\beta}_4) & O & O \\ O & diag(\boldsymbol{\beta}_5) & O & diag(\boldsymbol{\beta}_5) & O \\ -diag(\boldsymbol{\beta}_6) & diag(\boldsymbol{\beta}_6) & O & O & diag(\boldsymbol{\beta}_6) \end{pmatrix}$$

Then its dual formula can be written as

$$\begin{array}{ll} Z_{CD1} = \min & \alpha_0 + \boldsymbol{\mu}_d^\top \boldsymbol{\beta}_0 + \Sigma_d \bullet \Gamma_0 + \mathbf{1}^\top \boldsymbol{\beta_1} + \mathbf{1}^\top \boldsymbol{\beta_2} + \mathbf{1}^\top \boldsymbol{\beta_4} + \mathbf{1}^\top \boldsymbol{\beta_5} + \mathbf{1}^\top \boldsymbol{\eta}^{(3)} + \mathbf{1}^\top \boldsymbol{\eta}^{(4)} + \mathbf{1}^\top (H \circ \Gamma)\mathbf{1} + \mathbf{1}^\top \boldsymbol{\mu}_\mathbf{d} \\ s.t. & \begin{pmatrix} \alpha_0 & \frac{1}{2}(\boldsymbol{\beta_0} + \boldsymbol{\eta}^{(1)} + \boldsymbol{\eta}^{(3)})^\top & \frac{1}{2}\mathbf{w}^\top & \frac{1}{2}(\boldsymbol{\eta}^{(2)} + \boldsymbol{\eta}^{(3)})^\top \\ \frac{1}{2}\boldsymbol{\beta_0} + \frac{1}{2}(\boldsymbol{\eta}^{(1)} + \boldsymbol{\eta}^{(3)}) & \Gamma_0 + diag(\boldsymbol{\eta}^{(4)} - \boldsymbol{\eta}^{(1)}) & O & diag(\boldsymbol{\eta}^{(4)}) \\ \frac{1}{2}\mathbf{w} & O & W & O \\ \frac{1}{2}(\boldsymbol{\eta}^{(2)} + \boldsymbol{\eta}^{(3)}) & diag(\boldsymbol{\eta}^{(4)}) & O & diag(\boldsymbol{\eta}^{(4)} - \boldsymbol{\eta}^{(2)}) \end{pmatrix} - \begin{pmatrix} 0 & \mathbf{0}^\top & \mathbf{0}^\top & O \\ \mathbf{0} & O & \frac{1}{2}I & O \\ \mathbf{0} & \frac{1}{2}I & O & O \\ \mathbf{0} & O & O & O \end{pmatrix} \succcurlyeq_{co} 0 \end{array} \tag{30}$$

(2) Conic Formulation in the Process Flexibility Problem

Define

$$\mathbf{a}_i = \begin{pmatrix} \mathbf{e^m}_i \\ \mathbf{0}_n \\ \mathbf{e^m}_i \\ \mathbf{0}_n \end{pmatrix} \ \mathbf{a}_{m+i} = \begin{pmatrix} \mathbf{0}_m \\ \mathbf{e^n}_i \\ \mathbf{0}_m \\ \mathbf{e^n}_i \end{pmatrix} \ \hat{\mathbf{c}} = \begin{pmatrix} \mathbf{0}_m \\ \mathbf{c} \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix}, \ \hat{I} = \begin{pmatrix} I_m \\ O_{n\times m} \\ O_{m\times m} \\ O_{n\times m} \end{pmatrix},$$

Then the completely positive program equivalent to the worst case expected value of (19) can be written as

$$
\begin{array}{llll}
Z_{C2} = \max & -\hat{\mathbf{c}}^\top \mathbf{p} - \hat{I} \bullet Y & & \\
s.t. & 1 - p_j - p_{m+i} + X_{m+i,j} & = 0, & \forall (i,j) \in \mathcal{A}_0 \\
& \mathbf{a}_i^\top \mathbf{p} & = 1, & \forall i \in \{1,...,m+n\} \\
& \mathbf{a}_i^\top X \mathbf{a}_i & = 1, & \forall i \in \{1,...,m+n\} \\
& X_{ii} & = p_i, & \forall i \in \{1,...,m+n\} \\
& \begin{pmatrix} 1 & \boldsymbol{\mu}_d^\top & \mathbf{p}^\top \\ \boldsymbol{\mu}_d & \Sigma_d & Y^\top \\ \mathbf{p} & Y & X \end{pmatrix} & \succcurlyeq_{cp} 0 &
\end{array}
\tag{31}
$$

Refine dual variable $\boldsymbol{\beta}^{(1)}$, $\boldsymbol{\beta}^{(2)}$ and $\boldsymbol{\beta}^{(3)}$ in (11) as follows:

$$
\boldsymbol{\beta}^{(1)} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \end{pmatrix} \; \boldsymbol{\beta}^{(2)} = \begin{pmatrix} \boldsymbol{\beta}_3 \\ \boldsymbol{\beta}_4 \end{pmatrix} \; \boldsymbol{\beta}^{(3)} = \begin{pmatrix} \boldsymbol{\beta}_5 \\ \boldsymbol{\beta}_6 \end{pmatrix}
$$

where $\boldsymbol{\beta}_1, \boldsymbol{\beta}_3, \boldsymbol{\beta}_5 \in \mathbb{R}^m$ and $\boldsymbol{\beta}_2, \boldsymbol{\beta}_4, \boldsymbol{\beta}_6 \in \mathbb{R}^n$. Define

$$
\mathbf{w} = \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \\ \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}_5 \\ \boldsymbol{\beta}_6 \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix} + \begin{pmatrix} (H \circ \Gamma)^\top \mathbf{1}_n \\ (H \circ \Gamma) \mathbf{1}_m \\ \mathbf{0}_m \\ \mathbf{0}_n \end{pmatrix}
$$

$$
W = \begin{pmatrix} diag(\boldsymbol{\beta}_3 + \boldsymbol{\beta}_5) & -\frac{1}{2}(H \circ \Gamma)^\top & diag(\boldsymbol{\beta}_3) & O \\ -\frac{1}{2}(H \circ \Gamma) & diag(\boldsymbol{\beta}_4 + \boldsymbol{\beta}_6) & O & diag(\boldsymbol{\beta}_4) \\ diag(\boldsymbol{\beta}_3) & O & diag(\boldsymbol{\beta}_3) & O \\ O & diag(\boldsymbol{\beta}_4) & O & diag(\boldsymbol{\beta}_4) \end{pmatrix}
$$

Then its dual formula can be written as

$$
\begin{array}{ll}
Z_{CD2} = \min & \alpha_0 + \boldsymbol{\mu}^\top \boldsymbol{\beta}_0 + \Sigma \bullet \Gamma_0 + \mathbf{1}_m^\top (\boldsymbol{\beta}_1 + \boldsymbol{\beta}_2) + \mathbf{1}_n^\top (\boldsymbol{\beta}_3 + \boldsymbol{\beta}_4) + \mathbf{1}_n^\top (H \circ \Gamma) \mathbf{1}_m \\
s.t. & \begin{pmatrix} \alpha_0 & \frac{1}{2}\boldsymbol{\beta}_{\mathbf{0}}^\top & \frac{1}{2}\mathbf{w}^\top \\ \frac{1}{2}\boldsymbol{\beta}_{\mathbf{0}} & \Gamma_0 & O \\ \frac{1}{2}\mathbf{w} & O & W \end{pmatrix} + \begin{pmatrix} 0 & \mathbf{0}^\top & \frac{1}{2}\hat{\mathbf{c}}^\top \\ \mathbf{0} & O & \frac{1}{2}\hat{I} \\ \frac{1}{2}\hat{\mathbf{c}} & \frac{1}{2}\hat{I} & O \end{pmatrix} \succcurlyeq_{co} 0
\end{array}
\tag{32}
$$