

Submitted to *Operations Research*
manuscript (Please, provide the manuscript number!)

Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.

Worst-case Analysis of Process Flexibility Designs

David Simchi-Levi

Engineering Systems Division, Department of Civil & Environmental Engineering, and Operations Research Center,
Massachusetts Institute of Technology, dslevi@mit.edu

Yehua Wei

Fuqua School of Business, Duke University, yehua.wei@duke.edu

Theoretical studies of process flexibility designs have mostly focused on expected sales. In this paper, we take a different approach by studying process flexibility designs from the worst-case point of view. To study the worst-case performances, we introduce the plant cover indices (PCIs), defined by bottlenecks in flexibility designs containing a fixed number of products. We prove that given a flexibility design, a general class of worst-case performance measures can be expressed as functions of the design's PCIs and the given uncertainty set. This result has several major implications. First, it suggests a method to compare the worst-case performances of different flexibility designs without the need to know the specifics of the uncertainty sets. Second, we prove that under symmetric uncertainty sets and a large class of worst-case performance measures, the long-chain, a celebrated sparse design, is superior to a large class of sparse flexibility designs including any design that has a degree of two on each of its product nodes. Third, we show that under stochastic demand, the classical Jordan and Graves (JG) index can be expressed as a function of the PCIs. Furthermore, the PCIs motivate a modified JG index that is shown to be more effective in our numerical study. Finally, the PCIs lead to a heuristic for finding sparse flexibility designs that perform well under expected sales and have lower risk measures in our computational study.

Key words: Process flexibility, flexible production, capacity planning, robust optimization, worst-case analysis

1. Introduction

Fierce competitions in today's global markets have led manufacturers to expand product portfolio in order to maintain market shares. Unfortunately, the increase in product offerings increases demand volatility and reduces forecast accuracy. This, coupled with a significant increase in volatility of commodity prices, forces manufacturers to look for new operations strategies to better match available supply with variable demand (Simchi-Levi (2010)). Consequently, many manufacturers

have started to adopt an operations strategy known as process flexibility, which is defined as the ability to “build different types of products in the same manufacturing plant or on the same production line at the same time” (Jordan and Graves (1995)).

Process flexibility has been applied in various industries, from the automotive to the consumer packaged goods industry (see Simchi-Levi (2010)), to better respond to market changes without significantly increasing operational cost, inventory levels, or response time. While effective, process flexibility does not come free. Indeed, in full (process) flexibility, each plant is capable of producing all product families and as a result, such a strategy requires a significant investment. Therefore, most firms are only willing to implement sparse or limited flexibility designs, that is, designs where each plant can produce only a few different products. Interestingly, it has been observed by both practitioners and academics that, in many situations, the effectiveness of certain sparse flexibility designs is almost the same as that of the full flexibility design (see Chou et al. (2008)).

These observations have motivated recent analytical work, e.g., Chou et al. (2010), Simchi-Levi and Wei (2012), in attempting to explain the effectiveness of a certain class of sparse flexibility designs. The objective of these papers is to compare *expected* demand satisfied under sparse flexibility design to that of full flexibility, under *stochastic* demand. In this paper, we take a different approach by studying the *worst-case*, also referred to as *robust*, point of view. That is, we model the unknown demand using an uncertainty set, and study the worst-case performance of a flexibility design among all the demand instances in the given uncertainty set.

The rest of the paper is organized as follows. In the remainder of this section, we introduce notation (Section 1.1) and a class of worst-case performance measures for flexibility designs (Section 1.2). This class includes the minimum demand satisfied by the design; the minimum ratio of the demand satisfied by the design and the demand satisfied by full flexibility; and, the largest absolute gap between the demand satisfied by full flexibility and that of the specific design under consideration. In what follows, we refer to these measures as the *robust measures* associated with a given design.

In Section 2, we provide a literature review. In Section 3, we introduce the *plant cover index* (PCI) and establish the connection between the PCIs and worst-case performance measures of flexibility designs. We apply the index to show that one flexibility design always performs better in worst-case than another if and only if the PCIs of the latter design are dominated by the PCIs of the former. In Section 4, we compare the worst-case performances of different sparse process flexibility designs under symmetric uncertainty sets. In particular, we prove that an important flexibility design called the *long chain* always has better worst-case performance than a class of sparse flexibility designs that includes any design where each product is produced by exactly two plants. In Section 5, we show that the classical Jordan and Graves (JG) index can be calculated

as a function of the PCIs. Combining insights from the JG index and the PCIs, we propose a new index for comparing flexibility designs that performs well in our computational study. Finally, in Section 6, we propose a class of heuristics for identifying sparse flexibility designs that perform well under expected sales and various risk measures.

1.1. Notation

Let \mathbb{R}^n denote the n -dimensional vector space of reals. In the paper, bold letters are reserved for vectors and matrices. For example, $\mathbf{x} \in \mathbb{R}^n$ is a vector with entries x_1, x_2, \dots, x_n . Also, we let $\min^i(\mathbf{x})$ denote the i -th smallest element in the set $\{x_1, x_2, \dots, x_n\}$. Finally, we let $[n]$ denote the set of integers from 1 to n and $\Sigma([n])$ denote the set of all permutations of $[n]$.

In this paper, we consider a system with m plants and n products for some arbitrarily fixed positive integers m and n . We let $A := \{a_1, a_2, \dots, a_m\}$ represent the set of plant nodes, and $B := \{b_1, b_2, \dots, b_n\}$ represent the set of product nodes. In our model, we assume plant i has a fixed capacity of c_i for $1 \leq i \leq m$.

A flexibility design \mathcal{A} is represented by a set of arcs that form a bipartite graph defined on sets A and B . For example, the *full flexibility* design is denoted by $\mathcal{F} := \{(a_i, b_j) | \forall 1 \leq i \leq m, 1 \leq j \leq n\}$. For any $u \in A \cup B$, define $N(u, \mathcal{A}) := \{v | (u, v) \text{ or } (v, u) \in \mathcal{A}\}$, that is, $N(u, \mathcal{A})$ is the set of neighbors of u in the bipartite graph defined by (A, B, \mathcal{A}) . Moreover, for set $S \subseteq A$ or $S \subseteq B$, we let $N(S, \mathcal{A}) := \cup_{u \in S} N(u, \mathcal{A})$. Throughout the paper, we will assume that $|N(u, \mathcal{A})| \geq 1$ for all $u \in A \cup B$; that is, we assume no flexibility design \mathcal{A} has isolated plant or product nodes. We say \mathcal{A} is *connected* if the undirected bipartite graph formed by \mathcal{A} is connected, i.e., $N(S, \mathcal{A}) \geq 1$, for any $S \subsetneq A \cup B$, $S \neq \emptyset$.

Given an instance of the demand vector \mathbf{d} , the total demand satisfied by a flexibility design \mathcal{A} , denoted by $P(\mathbf{d}, \mathcal{A})$, is defined as the objective value of the following linear program (LP):

$$P(\mathbf{d}, \mathcal{A}) := \max \sum_{(a_i, b_j) \in \mathcal{A}} f_{ij} \quad (1)$$

$$\text{s.t.} \quad \sum_{a_i \in N(b_j, \mathcal{A})} f_{ij} \leq d_j, \forall b_j \in B \quad (2)$$

$$\sum_{b_j \in N(a_i, \mathcal{A})} f_{ij} \leq c_i, \forall a_i \in A \quad (3)$$

$$f_{ij} \geq 0, \forall (a_i, b_j) \in \mathcal{A} \quad (4)$$

$$\mathbf{f} \in \mathbb{R}^{|\mathcal{A}|}. \quad (5)$$

We will refer to $P(\mathbf{d}, \mathcal{A})$ as *the sales of \mathcal{A} given \mathbf{d}* . It is easy to see that $P(\mathbf{d}, \mathcal{F})$ can be expressed as follows:

REMARK 1. $P(\mathbf{d}, \mathcal{F}) = \min\{\sum_{i=1}^m c_i, \sum_{i=1}^n d_i\}$.

Finally, a system is said to be balanced if $m = n$. In a balanced system, we define the *long chain*, denoted by \mathcal{C} , as $\mathcal{C} = \{(a_1, b_1), (a_2, b_2), \dots, (a_n, b_n)\} \cup \{(a_1, b_2), (a_2, b_3), \dots, (a_{n-1}, b_n), (a_n, b_1)\}$; and *dedicated design*, denoted by \mathcal{D} , as $\mathcal{D} = \{(a_i, b_i) | \forall 1 \leq i \leq n\}$ (see Figure 1). One can immediately see that $P(\mathbf{d}, \mathcal{D}) = \sum_{i=1}^n \min\{c_i, d_i\}$. Also, we say a design \mathcal{A} is a 2-flexibility design if any plant node and any product node is incident to exactly two arcs in \mathcal{A} .

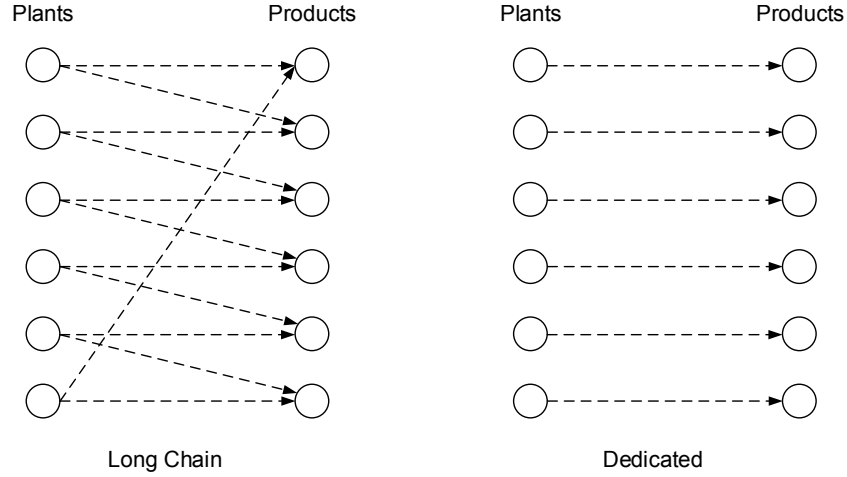


Figure 1 Designs for Balanced System with $n = 6$

1.2. Robust (Worst-case) Measures

A deterministic measure is a function that maps a demand instance $\mathbf{d} \in \mathbb{R}^n$ and a flexibility design \mathcal{A} to a real number. One example of such a function is the sales of a flexibility design, $P(\cdot)$. Given a deterministic measure function f , we use $R^f(\cdot)$ as the robust measure (or robust counterpart) of f , which is defined as

$$R^f(\mathcal{A}, U) := \min_{\mathbf{d} \in U} f(\mathbf{d}, \mathcal{A}).$$

In words, $R^f(\cdot)$ is a function that maps a flexibility design \mathcal{A} and an uncertainty set U to a real number, which measures the “robustness” of \mathcal{A} under U . Because the product demand is never negative, we will assume that any uncertainty set U considered in the paper lies in $(\mathbb{R}^+)^n$.

In this paper, we assume that any deterministic measure function f is continuous in \mathbf{d} . This assumption ensures that its robust counterpart, $R^f(\cdot)$, is always well defined. Next, we introduce three commonly used deterministic measure functions, denoted by f_s , f_r , and f_d , where

$$\begin{aligned} f_s(\mathbf{d}, \mathcal{A}) &:= P(\mathbf{d}, \mathcal{A}), \forall \mathbf{d} \in \mathbb{R}^n \\ f_r(\mathbf{d}, \mathcal{A}) &:= \frac{P(\mathbf{d}, \mathcal{A})}{P(\mathbf{d}, \mathcal{F})}, \forall \mathbf{d} \in \mathbb{R}^n \setminus \{\mathbf{0}\}, f_r(\mathbf{0}, \mathcal{A}) = 1, \\ f_d(\mathbf{d}, \mathcal{A}) &:= P(\mathbf{d}, \mathcal{A}) - P(\mathbf{d}, \mathcal{F}), \forall \mathbf{d} \in \mathbb{R}^n. \end{aligned}$$

To keep the notation simple, we let $R^s := R^{f_s}$, $R^r := R^{f_r}$ and $R^d := R^{f_d}$. Intuitively, R^s is the worst possible sales of design \mathcal{A} ; R^r is the worst possible ratio of the demand satisfied by \mathcal{A} to that of demand satisfied by full flexibility; and finally, R^d is the most negative gap between demand satisfied by full flexibility and demand satisfied by \mathcal{A} .

For any vector $\mathbf{d} \in \mathbb{R}^n$, define $\mathbf{d}^\sigma := [d_{\sigma(1)}, d_{\sigma(2)}, \dots, d_{\sigma(n)}]^T$ for any $\sigma \in \Sigma([n])$. We define $\Sigma(\mathbf{d}) := \{\mathbf{d}^\sigma | \forall \sigma \in \Sigma([n])\}$, that is, $\Sigma(\mathbf{d})$ is the set of all vectors that are permutations of \mathbf{d} . For any uncertainty set U , we say that U is *symmetric* if for any $\mathbf{d} \in U$, $\mathbf{d}^\sigma \in U$ for any permutation σ . Note that $\Sigma(\mathbf{d})$ is always symmetric, and if U is symmetric, then $\Sigma(\mathbf{d}) \subseteq U$ for any $\mathbf{d} \in U$.

In worst-case analysis, symmetric uncertainty sets are used for modeling symmetric demand variations. Some examples of symmetric uncertainty sets include

- triangle uncertainty, where $U = \{\sum_{i=1}^n d_i = t, d_i \geq 0, \forall 1 \leq i \leq n\}$ for some $t \in \mathbb{R}^+$;
- box uncertainty, where $U = \{l \leq d_i \leq u, \forall 1 \leq i \leq n\}$ for some $l, u \in \mathbb{R}^+$;
- ellipsoidal uncertainty, where $U = \{\sum_{i=1}^n (d_i - z)^2 \leq t, \forall 1 \leq i \leq n\}$ for some $z, t \in \mathbb{R}^+$;

or any intersection of the triangle, box or ellipsoidal uncertainty sets.

We note that the purpose of this paper is *not* to model demand uncertainties using an uncertainty set. Instead, we develop tools that can be applied to general classes of uncertainty sets (symmetric uncertainty sets in Section 3.2 and symmetric perturbation uncertainty sets in Section 3.3). Moreover, we develop results that identify flexibility designs performing well for not just one uncertainty set, but an entire class of uncertainty sets, and these results lead to design heuristics that are robust under a wide range of uncertainties.

We say that a deterministic measure function $f(\cdot)$ is *monotonic* in sales under fixed total demand if there exists a function g such that

$$f(\mathbf{d}, \mathcal{A}) = g(P(\mathbf{d}, \mathcal{A}), \sum_{i=1}^n d_i), \quad (6)$$

and $g(x, y)$ is strictly increasing in x with any fixed real number y .

In fact, most of the commonly used deterministic measure functions are monotonic in sales under fixed total demand. In particular, it is easy to check that $f_s(\cdot)$, $f_r(\cdot)$ and $f_d(\cdot)$ all satisfy this condition. Similarly, deterministic measure functions such as the capacity shortage, i.e., $f(\mathbf{d}, \mathcal{A}) = P(\mathbf{d}, \mathcal{A}) - \sum_{i=1}^n d_i$, and service rate, i.e., $f(\mathbf{d}, \mathcal{A}) = \frac{P(\mathbf{d}, \mathcal{A})}{\sum_{i=1}^n d_i}$ are also monotonic in sales under fixed total demand. We define Γ to be the set of all robust (worst-case) measures with deterministic measure functions that are monotonic in sales under fixed total demand.

2. Literature Review

Research on the effectiveness of sparse flexibility designs has first started with the seminal paper of Jordan and Graves (1995). The authors analyze a balanced manufacturing system where the

number of plants equals the number of products. They show, using empirical analysis, that a *long chain* design, a design in which all plants and products are connected in one cycle, performs almost as well as the full flexibility design from the average sales point of view. The authors then apply a similar concept, referred to as the *chaining strategy*, to unbalanced systems and show that even in this case, their design performs almost as well as full flexibility. Thus, their empirical work has two important implications. First, it suggests that a properly designed sparse flexibility can often capture all the benefits of full flexibility. Second, it provides a useful guideline on how to create effective sparse flexibility designs.

Following the work of Jordan and Graves, researchers have attempted to explain analytically the observed effectiveness of the long chain and other sparse flexibility designs. Aksin and Karaesmen (2007) show that there is a decrease in marginal benefit associated with the increase in either the degree of flexibility or the capacities of the manufacturing plants. Chou et al. (2010) develop a method to compute the average demand satisfied by the long chain in asymptotic regime. Using this method, they show that for some demand distributions, the average sales associated with the long chain is very close to that of full flexibility when the system size approaches infinity. Finally, the paper by Simchi-Levi and Wei (2012) identifies a decomposition for the expected demand satisfied by the long chain and applies the decomposition to prove several properties of the long chain for any finite system size. In particular, the paper proves that the long chain is optimal in average sales among all 2-flexibility designs, i.e., the degree for every plant and degree product is two, and derives a bound on the gap between the average sales of full flexibility and that of the long chain. Much like Jordan and Graves (1995), these research papers study flexibility designs under stochastic demand and focus on their average-case performances.

By contrast, very little research has focused on worst-case performance measures for flexibility designs. A rare exception is the work of Chou et al. (2011), which proves in a n plants and n products system, when the demand for each product is bounded by λ times the capacity of each plant, then an $(\alpha, \lambda, \Delta)$ -expander always performs within $(1 - \alpha\lambda)$ -optimality of the full flexibility design. Chou et al. (2011) also generalize the result to unbalanced systems, i.e., systems where the number of products is not equal to the number of plants, with non-homogenous plants and products. The main difference between Chou et al. (2011) and the current paper is that Chou et al. (2011) establishes conditions to identify sparse flexibility designs that are guaranteed to be within $(1 - \epsilon)$ -optimality of the full flexibility, whereas this paper establishes conditions and flexibility indices to compare the worst-case performances of different (sparse) flexibility designs.

An interesting question is whether one can compare the effectiveness of different flexibility designs without resorting to a detailed simulation study. To answer this question, the academic community started to develop flexibility design indices, following the original index developed in Jordan and

Graves (1995), which we will refer to as the JG index. Other well-known indices include the Structural Flexibility index in Iravani et al. (2005), WS-APL index in Iravani et al. (2007), g-Measure in Graves and Tomlin (2003), and the Expansion index in Chou et al. (2008). We refer the readers to Deng (2013), for a complete description of these indices.

Finally, following on the chaining strategy of Jordan and Graves (1995), different heuristics have also been proposed to generate effective sparse flexibility designs. Examples include the randomized sampling method of Chou et al. (2010), the node expansion method of Chou et al. (2011) and the unbalanced design guideline of Deng and Shen (2013).

3. Plant Cover Indices and Robust Measures

In order to develop some intuition, we start off the section (in Section 3.1 and 3.2) by assuming the uncertainty sets are symmetric. This assumption is relaxed in Section 3.3. For flexibility designs \mathcal{A}_1 and \mathcal{A}_2 , we say that \mathcal{A}_1 is *more symmetrically robust* than \mathcal{A}_2 if **for any** $R \in \Gamma$ and **any symmetric set** U , we have $R(\mathcal{A}_1, U) \geq R(\mathcal{A}_2, U)$. Moreover, we say \mathcal{A}_1 is *strictly more symmetrically robust* than \mathcal{A}_2 if \mathcal{A}_1 is more symmetrically robust than \mathcal{A}_2 , and there exists some symmetric set U and $R \in \Gamma$ that $R(\mathcal{A}_1, U) > R(\mathcal{A}_2, U)$. One can think of \mathcal{A}_1 being strictly more symmetrically robust than \mathcal{A}_2 similar to \mathcal{A}_2 being ‘‘Pareto dominated’’ by \mathcal{A}_1 in worst-case metrics.

3.1. Definition of the Plant Cover Indices

For any k , $0 \leq k \leq n$, we define *the plant cover index (PCI) at k* for flexibility design \mathcal{A} as the minimum plant capacities required to create a vertex cover on \mathcal{A} , given that the vertex cover contains exactly k products. The PCI at k , denoted by $\delta^k(\mathcal{A})$, is defined as the objective value of the following integer program:

$$\begin{aligned} \delta^k(\mathcal{A}) &:= \min \sum_{i=1}^m c_i p_i \\ \text{s.t.} \quad &\sum_{j=1}^n q_j = k, \\ &p_i + q_j \geq 1, \forall (a_i, b_j) \in \mathcal{A} \\ &\mathbf{p} \in \{0, 1\}^m, \mathbf{q} \in \{0, 1\}^n. \end{aligned}$$

It is straightforward to check that $\delta^0(\mathcal{A}) = \sum_{i=1}^m c_i$ and $\delta^n(\mathcal{A}) = 0$ for any \mathcal{A} . Therefore, different designs can only differ in $\delta^k(\cdot)$ for $1 \leq k < n$. Note that for any subset of products $S \subseteq B$, to create a vertex cover with $S \subseteq B$ and $S' \subseteq A$, one would need S' to cover all arcs that are incident to $N(B \setminus S, \mathcal{A})$. Therefore, $\delta^k(\mathcal{A})$ can be expressed as follows.

REMARK 2.

$$\delta^k(\mathcal{A}) = \min_{S \subseteq B, |S|=k} \sum_{a_i \in N(B \setminus S, \mathcal{A})} c_i.$$

One can think of Remark 2 as a *combinatorial* interpretation of $\delta^k(\mathcal{A})$. To develop more intuition for the PCI, recall that for any flexibility design \mathcal{A} and a fixed demand vector \mathbf{d} , $P(\mathbf{d}, \mathcal{A})$ is defined by the LP under Equations (1-5). The LP is actually a max-flow problem and from the classical max-flow min-cut theorem, we have

$$P(\mathbf{d}, \mathcal{A}) = \min \sum_{i=1}^m c_i p_i + \sum_{j=1}^n q_j d_j \quad (7)$$

$$\text{s.t. } p_i + q_j \geq 1, \forall (a_i, b_j) \in \mathcal{A} \quad (8)$$

$$\mathbf{p} \in \{0, 1\}^m, \mathbf{q} \in \{0, 1\}^n. \quad (9)$$

For any $S \subseteq B$, let $p_i = 1, \forall a_i \in N(B \setminus S, \mathcal{A})$, and $q_j = 1, \forall b_j \in S$, define

$$\mathcal{B}(S) := \sum_{a_i \in N(B \setminus S, \mathcal{A})} c_i + \sum_{b_j \in S} d_j.$$

Then, \mathbf{p}, \mathbf{q} are feasible for Equations (8) and (9), with objective value $\mathcal{B}(S)$. Therefore, $\mathcal{B}(S)$ is an upper bound on $P(\mathbf{d}, \mathcal{A})$. If $\mathcal{B}(S) < \sum_{i=1}^m c_i$, then S and $N(B \setminus S, \mathcal{A})$ form a bottleneck that blocks \mathcal{A} from utilizing all of the plant capacities. Note that $\mathcal{B}(S)$ is the sum of two quantities, $\sum_{a_i \in N(B \setminus S, \mathcal{A})} c_i$, the *total capacities of the bottleneck*, and $\sum_{b_j \in S} d_j$, the *total demands of the bottleneck*. Hence, by Remark 2, one can think of $\delta^k(\mathcal{A})$ as the minimum total capacities for a bottleneck containing exactly k products.

When demand is uncertain, there are exponentially many bottlenecks that may affect the sales of \mathcal{A} . However, in Section 3.2, we show that surprisingly, the capacities of the $n + 1$ bottlenecks corresponding to $(\delta^0(\mathcal{A}), \delta^1(\mathcal{A}), \dots, \delta^n(\mathcal{A}))$, form a *sufficient statistic* for determining the worst-case performance of \mathcal{A} under any fixed symmetric uncertainty set.

The PCIs defined in this section are related to two concepts in the process flexibility literature: the JG index in Jordan and Graves (1995), and the graph expanders in Chou et al. (2011). Here, we discuss the connection between the PCIs and the expanders. The connection between the PCIs and the JG index will be discussed later in Section 5.

In Chou et al. (2011), the authors stated that a design \mathcal{A} is an $(\alpha, \lambda, \Delta)$ -expander if (i) for every $u \in B$, $|N(u, \mathcal{A})| \leq \Delta$; and (ii) for any small subsets $S \subseteq B$ where $|S| \leq \alpha n$, we have $|N(S, \mathcal{A})| \geq \lambda |S|$. Therefore, under the assumption that all plants have unit capacity, a design \mathcal{A} is an $(\alpha, \lambda, \Delta)$ -expander if and only if

$$\min_{1 \leq k \leq \alpha n} \frac{\delta^{n-k}(\mathcal{A})}{k} \geq \lambda, \text{ and } |N(u, \mathcal{A})| \leq \Delta, \forall u \in B.$$

This connection illustrates that under the setting where all plants have unit capacity, one can use the PCIs to check if \mathcal{A} is an $(\alpha, \lambda, \Delta)$ -expander. Moreover, we note that while both the expander property and the PCIs (in Section 3.3) extend to more general demand uncertainties, a direct connection between the extensions does not exist.

3.2. Worst-case Measures with Symmetric Uncertainty Sets

To study the robust measures of \mathcal{A} under symmetric uncertainty sets, we first start with a lemma regarding $R^s(\mathcal{A}, U)$. Recall that R^s is the worst possible sales of \mathcal{A} under U , i.e., $R^s(\mathcal{A}, U) := \min_{\mathbf{d} \in U} P(\mathbf{d}, \mathcal{A})$.

LEMMA 1. *For any fixed $\mathbf{d} \in U$ and any integer $0 \leq k \leq n$,*

$$R^s(\mathcal{A}, U) \leq \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\mathbf{d}).$$

Proof. By definition of $\delta^k(\mathcal{A})$, we can find vectors $\mathbf{p}' \in \{0, 1\}^m$, $\mathbf{q}' \in \{0, 1\}^n$ such that $\sum_{i=1}^m c_i p'_i = \delta^k(\mathcal{A})$, $\sum_{j=1}^n q'_j = k$ and \mathbf{p}' , \mathbf{q}' are feasible for the optimization problem defined by Equation (7-9). Let σ be a permutation in $\Sigma([n])$ such that $q'_j = 1$ if and only if $d_{\sigma(j)} \in \{\min^i(\mathbf{d}) | 1 \leq i \leq k\}$. Then, we have that

$$\sum_{i=1}^m c_i p'_i + \sum_{j=1}^n q'_j d_{\sigma(j)} = \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\mathbf{d}).$$

Therefore, $P(\mathbf{d}^\sigma, \mathcal{A}) \leq \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\mathbf{d})$. Because U is symmetric, $\mathbf{d}^\sigma \in S$, which implies $R^s(\mathcal{A}, U) \leq P(\mathbf{d}^\sigma, \mathcal{A}) \leq \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\mathbf{d})$. \square

Next, we show that there always exists some integer k and vector \mathbf{d} such that the inequality in Lemma 1 is tight.

PROPOSITION 1. *Let $\boldsymbol{\tau} = \arg \min_{\mathbf{d} \in S} P(\mathbf{d}, \mathcal{A})$. Then,*

$$R^s(\mathcal{A}, U) = \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\boldsymbol{\tau})$$

for some nonnegative integer $0 \leq k \leq n$.

Proof. By the max-flow min-cut theorem, we have

$$\begin{aligned} P(\boldsymbol{\tau}, \mathcal{A}) &= \min \sum_{i=1}^m c_i p_i + \sum_{j=1}^n q_j \tau_j \\ \text{s.t.} \quad & p_i + q_j \geq 1, \forall (a_i, b_j) \in \mathcal{A} \\ & \mathbf{p} \in \{0, 1\}^m, \mathbf{q} \in \{0, 1\}^n. \end{aligned}$$

Let \mathbf{p}^* , \mathbf{q}^* be the optimal solution to the optimization problem above, and let $k := \sum_{j=1}^n q_j^*$. Then, we must have $\sum_{j=1}^n q_j^* \tau_j \geq \sum_{j=1}^k \min^j(\boldsymbol{\tau})$ and $\sum_{i=1}^m c_i p_i^* \geq \delta^k(\mathcal{A})$. Hence, we have that

$$R^s(\mathcal{A}, U) = P(\boldsymbol{\tau}, \mathcal{A}) = \sum_{i=1}^m c_i p_i^* + \sum_{j=1}^n q_j^* \tau_j \geq \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\boldsymbol{\tau})$$

But by Lemma 1, $R^s(\mathcal{A}, U) \leq \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\boldsymbol{\tau})$, and hence, we have $R^s(\mathcal{A}, U) = \delta^k(\mathcal{A}) + \sum_{i=1}^k \min^i(\boldsymbol{\tau})$. \square

From Lemma 1 and Proposition 1, we get

$$R^s(\mathcal{A}, U) = \min_{0 \leq k \leq n} \left\{ \delta^k(\mathcal{A}) + \min_{\mathbf{d} \in U} \sum_{i=1}^k \min^i(\mathbf{d}) \right\}.$$

The symmetric property of U implies that $\min_{\mathbf{d} \in U} \sum_{i=1}^k \min^i(\mathbf{d}) = \min_{\mathbf{d} \in U} \sum_{i=1}^k d_i$. Thus,

$$R^s(\mathcal{A}, U) = \min_{0 \leq k \leq n, \mathbf{d} \in U} \left\{ \delta^k(\mathcal{A}) + \sum_{i=1}^k d_i \right\}. \quad (10)$$

Equation (10) provides an explicit representation of $R^s(\mathcal{A}, U)$. Next, we generalize Equation (10) to any robust measure R that lies in Γ .

THEOREM 1. *Let f be a deterministic measure function that is monotonic in sales under fixed total demand. And let $g(\cdot)$ be the function such that $g(x, y)$ is strictly increasing in x for fixed y , and $f(\mathbf{d}, \mathcal{A}) = g(P(\mathbf{d}, \mathcal{A}), \sum_{i=1}^n d_i)$. Then,*

$$R^f(\mathcal{A}, U) = \min_{0 \leq k \leq n, \mathbf{d} \in U} \left\{ g(\delta^k(\mathcal{A}) + \sum_{i=1}^k d_i, \sum_{i=1}^n d_i) \right\}. \quad (11)$$

Proof. For any $T \in \mathbb{R}^+$, let $U_T := \{\mathbf{d} \mid \sum_{i=1}^n d_i = T\}$. Note that both U_T and $U_T \cap U$ are symmetric. Now we have

$$\begin{aligned} R^f(\mathcal{A}, U) &= \min_{\mathbf{d} \in U} \left\{ g(P(\mathbf{d}, \mathcal{A}), \sum_{i=1}^n d_i) \right\} \\ &= \min_{T \in \mathbb{R}^+} \left\{ \min_{\mathbf{d} \in U \cap U_T} \left\{ g(P(\mathbf{d}, \mathcal{A}), \sum_{i=1}^n d_i) \right\} \right\} \\ &= \min_{T \in \mathbb{R}^+} \left\{ \min_{\mathbf{d} \in U \cap U_T} \left\{ g(P(\mathbf{d}, \mathcal{A}), T) \right\} \right\} \\ \text{By property of } g(\cdot), &= \min_{T \in \mathbb{R}^+} \left\{ g\left(\min_{\mathbf{d} \in U \cap U_T} P(\mathbf{d}, \mathcal{A}), T \right) \right\} \\ \text{By Proposition 1,} &= \min_{T \in \mathbb{R}^+} \left\{ g\left(\min_{0 \leq k \leq n, \mathbf{d} \in U \cap U_T} \left\{ \delta^k(\mathcal{A}) + \sum_{i=1}^k d_i \right\}, T \right) \right\} \\ &= \min_{T \in \mathbb{R}^+, 0 \leq k \leq n, \mathbf{d} \in U \cap U_T} \left\{ g\left(\delta^k(\mathcal{A}) + \sum_{i=1}^k d_i, \sum_{i=1}^n d_i \right) \right\} \\ &= \min_{0 \leq k \leq n, \mathbf{d} \in U} \left\{ g\left(\delta^k(\mathcal{A}) + \sum_{i=1}^k d_i, \sum_{i=1}^n d_i \right) \right\}. \end{aligned}$$

\square

Theorem 1 thus shows that for any $R \in \Gamma$ and any symmetric uncertainty set U , there exists some function $H(\cdot)$ such that

$$R(\mathcal{A}, U) = H(\delta^0(\mathcal{A}), \delta^1(\mathcal{A}), \dots, \delta^n(\mathcal{A}), U). \quad (12)$$

Therefore, Equation (12) implies that if the values of $\delta^k(\mathcal{A})$ are given for all $0 \leq k \leq n$, then one can evaluate $R(\mathcal{A}, U)$, *without any additional information on \mathcal{A}* .

Theorem 1 provides a potentially practical algorithm for computing $R^f(\mathcal{A}, U)$ when $\delta^k(\mathcal{A})$ is given for $1 \leq k \leq n$. For example, suppose U is a symmetric polytope, and $f(\mathbf{d}, \mathcal{A}) = P(\mathbf{d}, \mathcal{A}) - P(\mathbf{d}, \mathcal{F})$, i.e. $R^f = R^d$. Then, let $g(x, y) = x - \min\{\sum_{i=1}^m c_i, y\}$, and we have $f(\mathbf{d}, \mathcal{A}) = g(P(\mathbf{d}, \mathcal{A}), \sum_{i=1}^n d_i)$. Applying Theorem 1, we obtain

$$R^d(\mathcal{A}, U) = \min_{0 \leq k \leq n, \mathbf{d} \in U} \left\{ \delta^k(\mathcal{A}) + \sum_{i=1}^k d_i - \min\left\{ \sum_{i=1}^m c_i, \sum_{i=1}^n d_i \right\} \right\} \quad (13)$$

$$= \min_{0 \leq k \leq n} \left\{ \delta^k(\mathcal{A}) + \min_{\mathbf{d} \in U} \left\{ \max\left\{ \sum_{i=1}^k d_i - \sum_{i=1}^m c_i, -\sum_{i=k+1}^n d_i \right\} \right\} \right\}, \quad (14)$$

where $\min_{\mathbf{d} \in U} \left\{ \max\left\{ \sum_{i=1}^k d_i - \sum_{i=1}^m c_i, -\sum_{i=k+1}^n d_i \right\} \right\}$ for each k can be computed by solving a LP. Therefore, $R^f(\mathcal{A}, U)$ can be computed by computing $n+1$ LPs. Of course, $\delta^k(\mathcal{A})$ for $1 \leq k \leq n$ is not always given, and determining them may not be an easy task. In Section 3.4, we will discuss in detail the computational complexity of determining $\delta^k(\mathcal{A})$, as well as our experience in computing $\delta^k(\mathcal{A})$ from numerical studies.

Another implication of Theorem 1 is a partial order of \mathcal{A}_1 and \mathcal{A}_2 under any worst-case measure in Γ . In particular, we have the following results.

THEOREM 2. *Fix a robust measure $R \in \Gamma$, then,*

$$R(\mathcal{A}_1, U) \geq R(\mathcal{A}_2, U), \text{ for any symmetric set } U,$$

*if and only if $\delta^k(\mathcal{A}_1) \geq \delta^k(\mathcal{A}_2)$ for $0 \leq k \leq n$.*¹

Proof. Let f be the deterministic measure function of R , and $g(\cdot)$ be the function such that $g(x, y)$ is strictly increasing in x for fixed y , and $f(\mathbf{d}, \mathcal{A}_t) = g(P(\mathbf{d}, \mathcal{A}_t), \sum_{i=1}^n d_i)$, for $t = 1, 2$. By Theorem 1, we have

$$R(\mathcal{A}_t, U) = \min_{0 \leq k \leq n, \mathbf{d} \in U} \left\{ g\left(\delta^k(\mathcal{A}_t) + \sum_{i=1}^k d_i, \sum_{i=1}^n d_i\right) \right\}, \forall t = 1, 2. \quad (15)$$

If $\delta^k(\mathcal{A}_1) \geq \delta^k(\mathcal{A}_2)$ for all k , $0 \leq k \leq n$, by Equation (15), we immediately obtain $R(\mathcal{A}_1, U) \geq R(\mathcal{A}_2, U)$.

Conversely, if $\delta^{k^*}(\mathcal{A}_1) < \delta^{k^*}(\mathcal{A}_2)$ for some k^* , let $C := \sum_{i=1}^m c_i$ and \mathbf{d}^* be the vector such that $d_i^* = 0$ for $1 \leq i \leq k^*$, and $d_i^* = C$ for $k^* < i \leq n$. Let $U^* := \Sigma(\mathbf{d}^*)$, for any \mathcal{A} , because $C \geq \delta^k(\mathcal{A})$ and $\delta^k(\mathcal{A}) \geq \delta^{k+1}(\mathcal{A})$ for any $0 \leq k < n$, we have

$$R(\mathcal{A}, U^*) = g(\delta^{k^*}(\mathcal{A}), (n - k^*)C).$$

Thus, we get $R(\mathcal{A}_1, U^*) < R(\mathcal{A}_2, U^*)$ as $g(x, y)$ is strictly increasing in x for fixed y . \square

An interesting question is whether better worst-case performance implies better average-case performance. Specifically, when \mathcal{A}_1 is strictly more symmetrically robust than \mathcal{A}_2 , we know that the worst-case performance of \mathcal{A}_1 is always better (and sometimes strictly better) than the worst-case performance of \mathcal{A}_2 . Would this imply that the expected sales of \mathcal{A}_1 is greater than or equal to the expected sales of \mathcal{A}_2 under any independent and identically distributed (IID) product demands? While our computational experiments suggest that a strictly more symmetrically robust flexibility design has higher expected sales in almost all of the randomly generated instances, the claim is not correct in general. This is proved by a counterexample in Appendix B.1.

3.3. PCI with Symmetric Perturbation Uncertainty Sets

The results in this section so far assume the symmetry of the uncertainty sets. In this subsection, we generalize our results to the following class of asymmetric uncertainty sets.

DEFINITION 1. A set U is *symmetric* around $\boldsymbol{\mu}$ if $E := \{\mathbf{x} - \boldsymbol{\mu} | \mathbf{x} \in U\}$ is symmetric. In this case, U is called a *symmetric perturbation uncertainty set*.

When U is the uncertainty set of the demand, one can interpret Definition 1 as having the demand for products estimated to be $\boldsymbol{\mu}$, while the perturbation (or error) of the estimation has the same fluctuation across products. An analogous scenario under stochastic demand is when the stochastic product demand vector is \mathbf{D} , $\boldsymbol{\mu} = \mathbb{E}[\mathbf{D}]$, and $\mathbf{D} - \boldsymbol{\mu}$ is an exchangeable (or IID) random vector. In that case, the set of samples of \mathbf{D} would appear like a symmetric set around $\boldsymbol{\mu}$, provided that the sample size is large. Also, note that U is *symmetric* if and only if U is *symmetric* around some $\boldsymbol{\mu}$, with $\mu_i = \mu_j$ for all $1 \leq i, j \leq n$.

For the rest of this subsection, we restrict our attention to symmetric perturbation uncertainty sets that are symmetric around some fixed $\boldsymbol{\mu}$, and let $E := \{\mathbf{d} - \boldsymbol{\mu} | \mathbf{d} \in U\}$. For the fixed $\boldsymbol{\mu}$, the PCIs are defined as

$$\begin{aligned} \delta_{\boldsymbol{\mu}}^k(\mathcal{A}) &:= \min \sum_{i=1}^m c_i p_i + \sum_{j=1}^n \mu_j q_j \\ \text{s.t.} \quad &\sum_{j=1}^n q_j = k, \\ &p_i + q_j \geq 1, \forall (a_i, b_j) \in \mathcal{A} \\ &\mathbf{p} \in \{0, 1\}^m, \mathbf{q} \in \{0, 1\}^n. \end{aligned}$$

Similar to Remark 2, $\delta_{\mu}^k(\mathcal{A})$ has the following combinatorial interpretation:

REMARK 3.

$$\delta_{\mu}^k(\mathcal{A}) = \min_{S \subseteq B, |S|=k} \sum_{a_i \in N(B \setminus S, \mathcal{A})} c_i + \sum_{b_j \in S} \mu_j.$$

In the rest of the paper, we will use $S^k \subset B$ to denote the set such that

$$\delta_{\mu}^k(\mathcal{A}) = \sum_{a_i \in N(B \setminus S^k, \mathcal{A})} c_i + \sum_{b_j \in S^k} \mu_j, \text{ and } |S^k| = k.$$

One can interpret $\delta_{\mu}^k(\mathcal{A})$ as the tightness of the (potential) bottleneck containing exactly k products, where a smaller $\delta_{\mu}^k(\mathcal{A})$ implies a tighter bottleneck. To see this, assume $\delta_{\mu}^k(\mathcal{A})$ is much greater than $\sum_{i=1}^m c_i$, and the uncertainty in the estimated demand is low, then intuitively, the bottleneck containing k products will not prevent \mathcal{A} from utilizing all of its plant capacities.

Next, we state two theorems analogous to Theorems 1 and 2. The proof for Theorem 3 is omitted due to its similarities with the proof for Theorem 1.

THEOREM 3. *Let f be the deterministic measure function that is monotonic in sales under fixed total demand. Let $g(\cdot)$ be the function such that $g(x, y)$ is strictly increasing in x for fixed y , and $f(\mathcal{A}, \mathbf{d}) = g(P(\mathcal{A}, \mathbf{d}), \sum_{i=1}^n d_i)$. Then for U that is symmetric around μ ,*

$$R^f(\mathcal{A}, U) = \min_{0 \leq k \leq n, \epsilon \in E} \left\{ g(\delta_{\mu}^k(\mathcal{A}) + \sum_{i=1}^k \epsilon_i, \sum_{i=1}^n (\mu_i + \epsilon_i)) \right\}, \quad (16)$$

where $E := \{\mathbf{d} - \mu \mid \mathbf{d} \in U\}$.

THEOREM 4. *Fix a robust measure $R \in \Gamma$, then,*

$$R(\mathcal{A}_1, U) \geq R(\mathcal{A}_2, U), \text{ for any } U \text{ that is symmetric around } \mu,$$

if and only if $\delta_{\mu}^k(\mathcal{A}_1) \geq \delta_{\mu}^k(\mathcal{A}_2)$ for $0 \leq k \leq n$.

Proof. Let $g(\cdot)$ be the function such that $g(x, y)$ is strictly increasing in x for fixed y , and $R(\mathcal{A}, U) = \min_{\mathbf{d} \in U} \{g(P(\mathbf{d}, \mathcal{A}), \sum_{i=1}^n d_i)\}$. By Theorem 3, we have

$$R(\mathcal{A}, U) = \min_{0 \leq k \leq n, \epsilon \in E} \left\{ g(\delta_{\mu}^k(\mathcal{A}) + \sum_{i=1}^k \epsilon_i, \sum_{i=1}^n (\mu_i + \epsilon_i)) \right\}.$$

If $\delta_{\mu}^k(\mathcal{A}_1) \geq \delta_{\mu}^k(\mathcal{A}_2)$ for all k , $0 \leq k \leq n$, we immediately get that $R(\mathcal{A}_1, U) \geq R(\mathcal{A}_2, U)$.

Conversely, if $\delta_{\mu}^{k^*}(\mathcal{A}_1) < \delta_{\mu}^{k^*}(\mathcal{A}_2)$ for some k^* , let K be a constant such that $|\delta_{\mu}^{k_1}(\mathcal{A}_2) - \delta_{\mu}^{k_2}(\mathcal{A}_2)| < K$ for any $0 \leq k_1 \neq k_2 \leq n$. Also let \mathbf{e}^* be the vector such that $e_i^* = -K$ for $1 \leq i \leq k^*$, and $e_i^* = K$ for $k^* < i \leq n$. Let $U^* := \mu + \Sigma(\mathbf{e}^*)$, and because of our choice of U^* , we have

$$R(\mathcal{A}_2, U^*) = g(\delta_{\mu}^{k^*}(\mathcal{A}_2) - k^*K, \sum_{i=1}^n \mu_i + (n - 2k^*)K)$$

$$\begin{aligned}
&> g(\delta_{\mu}^{k^*}(\mathcal{A}_1) - k^*K, \sum_{i=1}^n \mu_i + (n - 2k^*)K) \\
&\geq R(\mathcal{A}_1, U^*).
\end{aligned}$$

□

We note that the notion of the PCIs can be extended even further, to a setting with linear production constraints. This extension is presented in Appendix C. Finally, we remark that computing $\delta^k(\mathcal{A})$ requires only the structure of \mathcal{A} and plant capacities, while computing $\delta_{\mu}^k(\mathcal{A})$ requires the structure of \mathcal{A} , plant capacities and the estimation of the product demands. Thus, the variations of the PCIs we proposed can adapt to different levels of demand information, to either the setting with no demand information, or the setting when demand has an estimated mean.

3.4. Hardness Result

In this subsection, we apply the connection we established between $\delta_{\mu}^k(\mathcal{A})$ and $R(\mathcal{A}, U)$, to prove a hardness result on computing $R(\mathcal{A}, U)$. In particular, we prove that for any $R \in \Gamma$, computing $R(\mathcal{A}, U)$ is an NP-hard problem.

To establish the hardness result, we begin with a lemma, which applies a result obtained by Kuo and Fuchs (1987), where the authors studied the problem of optimally reconfiguring processor arrays with faulty cells.

LEMMA 2. *Given non-negative integers k, t and some flexibility design \mathcal{A} , determining whether $\delta^k(\mathcal{A}) \leq t$ is NP-hard.*

Proof. Consider the case $c_i = 1$ for all $1 \leq i \leq m$. In this case, note that $\delta^k(\mathcal{A}) \leq t$ if and only if there is a vertex cover $V_A \cup V_B$, where $V_A \subseteq A$, $|V_A| \leq t$ and $V_B \subseteq B$, $|V_B| \leq k$. Kuo and Fuchs (1987) proved that it is NP-hard to determine if there exists such a vertex cover. Thus, we have that determining whether $\delta^k(\mathcal{A}) \leq t$ is NP-hard. □

We note that the problem of determining the existence of a vertex cover $V_A \cup V_B$, such that $V_A \subseteq A$, $|V_A| \leq t$ and $V_B \subseteq B$, $|V_B| \leq k$ is known in the computer science literature as the “constraint bipartite vertex cover” problem. Despite the problem being NP-hard, researchers have developed (exponential) algorithms to compute the constraint bipartite vertex cover (and hence $\delta^k(\mathcal{A})$) that work quite well in practice (see Fernau and Niedermeier (2001), Bai and Fernau (2008)).

Having established Lemma 2, we now prove that computing $R^f(\mathcal{A}, U)$ for any $R^f \in \Gamma$ is NP-hard.

COROLLARY 1. *Fix any robust measure $R^f \in \Gamma$, determining whether $R^f(\mathcal{A}, U) \leq t$ for symmetric U is NP-hard.*

Proof. We prove this result by showing that for $c_i = 1$ for all $1 \leq i \leq m$, the problem of determining if $\delta^k(\mathcal{A}) \leq t$ for some integer t can be reduced to the problem of determining if $R^f(\mathcal{A}, U) \leq t'$ for some $t' \in \mathbb{R}$ and $U \subseteq \mathbb{R}^n$.

We can assume $t < m$, as $\delta^k(\mathcal{A}) \leq m$. Because $R^f \in \Gamma$, we can find some function $g(x, y)$ that is strictly increasing in x for any fixed y , where $f(\mathcal{A}, \mathbf{d}) = g(P(\mathcal{A}, \mathbf{d}), \sum_{i=1}^n d_i)$.

Let \mathbf{d}^* be the vector such that $d_j^* = 0$ for $1 \leq j \leq k$ and $d_j^* = m$ for $k+1 \leq j \leq n$. Let $U = \Sigma(\mathbf{d}^*)$, because $\sum_{i=1}^n d_i = (n-k)m$ for any $\mathbf{d} \in U$, we have that

$$R^f(\mathcal{A}, U) = \min_{\mathbf{d} \in S} g(P(\mathcal{A}, \mathbf{d}), \sum_{i=1}^n d_i) = g(\min_{\mathbf{d} \in S} P(\mathcal{A}, \mathbf{d}), (n-k)m).$$

By the construction of U and Proposition 1, we have $\min_{\mathbf{d} \in U} P(\mathbf{d}, \mathcal{A}) = \delta^k(\mathcal{A})$. Thus, $R^f(\mathcal{A}, U) = g(\delta^k(\mathcal{A}), (n-k)m) \leq t' := g(t, (n-k)m)$ if and only if $\delta^k(\mathcal{A}) \leq t$. Therefore, we have that determining whether $R(\mathcal{A}, U) \leq t'$ is at least as hard as determining whether $\delta^k(\mathcal{A}) \leq t$. \square

We would like to point out that while Lemma 2 shows that computing $\delta^k(\mathcal{A})$ (and hence $\delta_{\mu}^k(\mathcal{A})$) is NP-hard, off-the-shelf solvers such as CPLEX are capable of determining $\delta_{\mu}^k(\mathcal{A})$ very quickly. In our computational experience, the binary program solver in CPLEX has consistently solved $\delta_{\mu}^k(\mathcal{A})$ for systems with 30 plant notes and 100 products within a second on a standard T430 Lenovo laptop. As a result, instead of studying theoretically efficient approximation algorithms for computing $\delta_{\mu}^k(\mathcal{A})$, we focus on applying the concept of $\delta_{\mu}^k(\mathcal{A})$ to identify effective flexibility designs.

4. Worst-case Performance of the Long Chain

In this section, we apply the results from the previous section to analyze the worst-case effectiveness of sparse flexibility designs. In particular, we are interested in the long chain design, \mathcal{C} , which has been studied extensively in the literature from the average-case point of view. As is typical in the analysis of the long chain, see for example Simchi-Levi and Wei (2012), we consider a balanced system, i.e. $m = n$, and use n to denote the number of plants and products. Also, we assume that the demand uncertainty set is symmetric and capacities are equal across the plants. Under this assumption, without loss of generality, we let $c_i = 1, \forall 1 \leq i \leq n$.

Consider the class of all flexibility designs in which each product is produced by exactly two plants. The theorem below shows that the long chain is more symmetrically robust than any other flexibility design in this class.

THEOREM 5. *Let \mathcal{A} be a design such that for any $u \in B$, $|N(u, \mathcal{A})| = 2$. Then, the long chain flexibility design, \mathcal{C} , is more symmetrically robust than \mathcal{A} . That is, for any symmetric set U and any $R \in \Gamma$, $R(\mathcal{C}, U) \geq R(\mathcal{A}, U)$.*

Proof. It is easy to check that $\delta^k(\mathcal{C}) = n - k + 1$ for $1 \leq k \leq n - 1$, and $\delta^n(\mathcal{C}) = 0 = \delta^n(\mathcal{A})$. To prove Theorem 5, it is sufficient to show that for all $1 \leq k < n$, we can find some $S \subset B$, $|S| = k$, such that $|N(B \setminus S, \mathcal{A})| \leq n - k + 1$, as $\delta^k(\mathcal{A}) \leq |N(B \setminus S, \mathcal{A})|$.

Suppose the graph formed by \mathcal{A} consists of c connected bipartite components. For $1 \leq i \leq c$, let $A_i \subset A$, $B_i \subset B$ be the set of vertices of the i -th component. Without loss of generality, we also assume that $|A_i| - |B_i|$ is non-decreasing with i . Because $\sum_{i=1}^c (|A_i| - |B_i|) = 0$, this assumption implies that $\sum_{i=1}^t |A_i| \leq \sum_{i=1}^t |B_i|$ for any $t \leq c$.

We now show that for any i , and any $1 \leq l \leq |B_i|$, there exists some $T \subseteq B_i$, $|T| = l$ such that $|N(T, \mathcal{A})| \leq l + 1$. This is done by induction on l . For $l = 1$, take any $u \in B_i$, take $T = \{u\}$ and $|N(T, \mathcal{A})| = 2$. Suppose the statement is true for some $l < |B_i|$, then we can find set $T^l \subseteq B_i$, $|T^l| = l$ and $|N(T^l, \mathcal{A})| \leq l + 1$. Since the vertices in $A_i \cup B_i$ form a connected component, and $T^l \subsetneq B_i$, there exists some $u \in N(T^l, \mathcal{A})$ such that (u, v) is an arc for some $v \notin T^l$. Since $|N(v, \mathcal{A})| = 2$ and $u \in N(T^l, \mathcal{A})$, we must have that $|N(T^l \cup \{v\}, \mathcal{A})| \leq l + 2$. Thus, by induction, we have that for any $1 \leq l \leq |B_i|$, there exists some $T \subseteq B_i$, $|T| = l$ such that $|N(T, \mathcal{A})| \leq l + 1$.

For any $1 \leq k < n$, let t_k the largest possible t such that $\sum_{i=1}^t |B_i| < n - k$. By our choice of t_k , we have $t_k < c$ and $n - k - \sum_{i=1}^{t_k} |B_i| \leq |B_{t_k+1}|$. Thus, we can find some set T where $|T| = n - k - \sum_{i=1}^{t_k} |B_i|$, $T \subseteq B_{t_k+1}$ and $|N(T, \mathcal{A})| \leq n - k - \sum_{i=1}^{t_k} |B_i| + 1$. Finally, let $S := (B_{t_k+1} \cup B_{t_k+2} \cup \dots \cup B_c) \setminus T$, and we have

$$|N(B \setminus S, \mathcal{A})| = |N(T, \mathcal{A})| + \sum_{i=1}^{t_k} |A_i| \leq n - k - \sum_{i=1}^{t_k} |B_i| + 1 + \sum_{i=1}^{t_k} |B_i| \leq n - k + 1.$$

Since $S \subset B$ and $|S| = n - \sum_{i=1}^{t_k} |B_i| - (n - k - \sum_{i=1}^{t_k} |B_i|) = k$, the proof is complete. \square

The result stated by Theorem 5 strongly favors the long chain, as it proves that under *any* robust measure $R \in \Gamma$ and *any* symmetric uncertainty set, long chain is always guaranteed to be optimal among all designs in which each product is produced by exactly two plants. Interestingly, Simchi-Levi and Wei (2012) prove that long chain is at least as good as any 2-flexibility designs under stochastic exchangeable demand.

Next, we present another result on the robustness of the long chain relative to that of a connected sparse flexibility design.

THEOREM 6. *The long chain flexibility design, \mathcal{C} , is more symmetrically robust than \mathcal{A} , if $|\mathcal{A}| = 2n$, and the bipartite graph with vertex sets A , B and arc set \mathcal{A} is connected.*

Proof. For $n = 1$, it is simple to check that Theorem 6 holds. Suppose \mathcal{A}^* is a counterexample to Theorem 6 in the *smallest* system (the smallest n^* where there is a counterexample). Because \mathcal{A} must be the same as \mathcal{C} for $n = 2$, we must have $n^* > 2$. Since \mathcal{A}^* is a counterexample, there

exists some $1 \leq k^* < n^*$ such that $\delta^{k^*}(\mathcal{A}^*) > n^* - k^* + 1$. By Theorem 5, we know there must exist some $u \in B$, with $|N(u, \mathcal{A}^*)| = 1$. Let $v = N(u, \mathcal{A}^*)$, and let G be the bipartite graph with vertex sets A, B , and arc set \mathcal{A}^* . Because G is connected, we must have $|N(v, \mathcal{A}^*)| \geq 2$.

Let $\mathcal{A}' = \{(v', u') | (v', u') \in \mathcal{A}^*, u' \neq u, v' \neq v\}$. Consider the bipartite graph G' with vertex sets $A \setminus v, B \setminus u$, and arc set \mathcal{A}' . Suppose G' has z components; then we must have $|N(v, \mathcal{A}^*)| \geq z + 1$. In this case, we can add $z - 1$ arcs to G' so that G' is a connected bipartite graph. Let \mathcal{A}'' be the arc set that contains \mathcal{A}' and the $z - 1$ added arcs. Note that $|\mathcal{A}''| \leq 2(n^* - 1)$.

By construction, the bipartite graph with vertex sets $A \setminus u, B \setminus v$ and arc set \mathcal{A}'' is connected. Because $1 \leq k^* < n^*$, the minimality assumption on \mathcal{A}^* and Remark 2, there exists some $S \subset B \setminus v$, with $|S| = n^* - k^* - 1$ and $|N(S, \mathcal{A}'')| \leq n^* - k^*$. But this implies that $S \cup \{v\} \subseteq B$, $|S \cup \{v\}| = n^* - k^*$ and $|N(S \cup \{v\}, \mathcal{A}'')| \leq n^* - k^* + 1$. By Remark 2, $|N(S \cup \{v\}, \mathcal{A}'')| \leq n^* - k^* + 1$ implies that $\delta^{k^*}(\mathcal{A}^*) \leq n^* - k^* + 1$. This contradicts our assumption that $\delta^{k^*}(\mathcal{A}^*) > n^* - k^* + 1$ and therefore, we have that Theorem 6 must be true. \square

A natural generalization to Theorems 5 and 6 is to compare the long chain to all other flexibility designs with $2n$ arcs. To our surprise, there exists a counterexample (see Appendix B.2) where for some worst-case performance measure, the long chain is inferior to another design with $2n$ arcs. We believe that not only is the counterexample important from the theoretical point of view, but that it also provides the following interesting intuition. In particular, in large systems, the long chain becomes less robust because the bottlenecks containing k products become very tight for some integer k . Moreover, these bottlenecks can be relaxed by just adding a few links to the long chain. Indeed, our counterexample leverages this fact by first creating a large chain with several isolated arcs, and then adding the last few remaining arcs to the large chain (see Appendix B.2). In this case, the design \mathcal{A} we construct has $\delta^k(\mathcal{A}) > \delta^k(\mathcal{C})$ for some $1 \leq k \leq n - 1$.

Observe that by Theorem 5, if a design \mathcal{A} with $2n$ arcs has $\delta^k(\mathcal{A}) > \delta^k(\mathcal{C})$ for some k , then there is some node $u \in B$ where $|N(u, \mathcal{A})| = 1$. But this implies that $\delta^{n-1}(\mathcal{A}) = 1 < 2 = \delta^{n-1}(\mathcal{C})$. Hence, there is no design with $2n$ arcs that is strictly more symmetrically robust than \mathcal{C} . That is, \mathcal{C} is in some sense a ‘‘Pareto optimal’’ design among all flexibility designs with $2n$ arcs in worst-case performances.

Finally, we prove that $\delta^k(\mathcal{A}) \leq \delta^k(\mathcal{C})$, when k is close to 0 or close to n . This result is formally stated as Proposition 2. We relegate the proof to Appendix A, due to its technical nature and relatively limited scope.

PROPOSITION 2. *In a balanced system with equal plant capacities, for any integer $0 \leq k \leq \alpha\sqrt{n}$, where $\alpha = 2 - \frac{2}{\sqrt{n}}$ we have*

$$\delta^{n-k}(\mathcal{C}) \geq \delta^{n-k}(\mathcal{A}) \text{ and } \delta^k(\mathcal{C}) \geq \delta^k(\mathcal{A}),$$

for any \mathcal{A} such that $|\mathcal{A}| = 2n$.

Proposition 2 indicates that under some uncertainty sets the long chain has better worst-case performance than any design \mathcal{A} with $2n$ arcs. For example, if the uncertainty set $U \subseteq L$, where

$$L := \{\mathbf{d} \mid \text{the total number of indices } i \text{ such that } d_i < 1 \text{ is less than } 2\sqrt{n} - 2\},$$

then we have $R(\mathcal{A}, U) \leq R(\mathcal{C}, U)$, for any $R \in \Gamma$. Intuitively, when $U \subseteq L$, we have that most of the product demands are greater or equal to the plant capacities (which is equal to 1), except for just a few products.

5. Evaluating Different Flexibilities Designs

In Section 3, we showed that for any $R \in \Gamma$ and any uncertainty set U that is symmetric with $\boldsymbol{\mu}$, we can always find some function $H(\cdot)$ such that

$$R(\mathcal{A}, U) = H(\delta_{\boldsymbol{\mu}}^0(\mathcal{A}), \delta_{\boldsymbol{\mu}}^1(\mathcal{A}), \dots, \delta_{\boldsymbol{\mu}}^n(\mathcal{A}), U).$$

Motivated by this observation, in this section, we attempt to use the PCIs as statistics to quickly evaluate flexibility designs. In particular, we want to find a function $I(\cdot)$, such that $I(\delta_{\boldsymbol{\mu}}^0(\mathcal{A}), \delta_{\boldsymbol{\mu}}^1(\mathcal{A}), \dots, \delta_{\boldsymbol{\mu}}^n(\mathcal{A}))$ allows us to estimate the effectiveness of \mathcal{A} under a given stochastic demand \mathbf{D} . Throughout the section, we assume that $\boldsymbol{\mu}$ is the expectation of \mathbf{D} . Also, we let $c := \sum_{i=1}^m c_i$, and $\mu := \sum_{j=1}^n \mu_j$.

5.1. JG Index

In this subsection, we prove that the classical JG index from Jordan and Graves (1995) can be viewed as a function of the PCIs. First, we introduce the formal definition of the JG index.

DEFINITION 2. The JG index of a flexibility design \mathcal{A} , denoted by $JG(\mathcal{A})$, is defined as

$$JG(\mathcal{A}) := \max_{S \subseteq B} \{\Pi(\mathcal{A}, S)\},$$

where

$$\begin{aligned} \Pi(\mathcal{A}, S) &:= \mathbb{P}\left[\sum_{b_j \in S} D_j - \sum_{a_i \in N(S, \mathcal{A})} c_i > \max\left(0, \sum_{j=1}^n D_j - \sum_{i=1}^m c_i\right)\right] \\ &= \mathbb{P}\left[\sum_{a_i \in N(S, \mathcal{A})} c_i < \sum_{b_j \in S} D_j, \sum_{b_j \in B \setminus S} D_j < \sum_{a_i \in A \setminus N(S, \mathcal{A})} c_i\right] \\ &= \mathbb{P}\left[\sum_{b_j \in B \setminus S} D_j + \sum_{a_i \in N(S, \mathcal{A})} c_i < \sum_{j=1}^n D_j, \sum_{b_j \in B \setminus S} D_j + \sum_{a_i \in N(S, \mathcal{A})} c_i < \sum_{i=1}^m c_i\right]. \end{aligned}$$

In words, $\Pi(\mathcal{A}, S)$ can be interpreted as the probability that the bottleneck formed by products in $B \setminus S$ and plants in $N(S, \mathcal{A})$ blocks \mathcal{A} from both *utilizing all of the plant capacities, and satisfying all of the customer demands*. Because the fully flexible system always either utilizes all plant capacities or satisfies all customer demands, $\Pi(\mathcal{A}, S)$ is also the probability that the bottleneck formed by $B \setminus S$ and $N(S, \mathcal{A})$ blocks \mathcal{A} from achieving the same sales as full flexibility. For convenience, $\Pi(\mathcal{A}, S)$ will sometimes be referred to as the *blocking probability* of the bottleneck formed by products in $B \setminus S$ and plants in $N(S, \mathcal{A})$, and therefore, $JG(\mathcal{A})$ is equal to the largest blocking probability achieved by a bottleneck. In Jordan and Graves (1995), the authors argue that if $JG(\mathcal{A})$ is low, then \mathcal{A} does not have any tight bottleneck, thus implying that \mathcal{A} is almost as effective as full flexibility.

The definition of the $JG(\mathcal{A})$ is similar to the combinatorial interpretation of $\delta_{\mu}^k(\mathcal{A})$. Indeed, it turns out that when demands are independent normals and the standard deviations of demands are equal across products, the JG index can be expressed as a function of $\delta_{\mu}^k(\mathcal{A})$, for $0 < k < n$, where μ is the mean of the product demands. This is stated formally in the next proposition.

PROPOSITION 3. *Suppose \mathbf{D} is an independent normal vector with mean μ , and there exists some σ such that $\mathbb{E}[(D_j - \mu)^2] = \sigma^2$ for any $1 \leq j \leq n$. Let $c := \sum_{i=1}^m c_i$ and $\mu := \sum_{j=1}^n \mu_j$; then*

$$JG(\mathcal{A}) = \max_{1 \leq k < n} (1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - c}{\sqrt{k}\sigma}))(1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - \mu}{\sqrt{n-k}\sigma})),$$

where Φ is the cumulative distribution function (CDF) of the standard normal distribution.

Proof. For each $S \subsetneq B$, rearranging the expression for $\Pi(\mathcal{A}, S)$, we get

$$\Pi(\mathcal{A}, S) = \mathbb{P}[\sum_{b_j \in S} D_j - \sum_{a_i \in N(S, \mathcal{A})} c_i > 0, \sum_{b_j \in B \setminus S} D_j - \sum_{a_i \in A \setminus N(S, \mathcal{A})} c_i < 0].$$

And because \mathbf{D} is independent, we have

$$\Pi(\mathcal{A}, S) = \mathbb{P}[\sum_{b_j \in S} D_j - \sum_{a_i \in N(S, \mathcal{A})} c_i > 0] \mathbb{P}[\sum_{b_j \in B \setminus S} D_j - c + \sum_{a_i \in N(S, \mathcal{A})} c_i < 0].$$

Note that for any $S \subsetneq B$, we have $\sum_{a_i \in N(S, \mathcal{A})} c_i + \sum_{b_j \in B \setminus S} \mu_j \geq \delta_{\mu}^{n-|S|}(\mathcal{A})$, and hence,

$$\begin{aligned} \Pi(\mathcal{A}, S) &= \mathbb{P}[\sum_{b_j \in S} D_j - \sum_{a_i \in N(S, \mathcal{A})} c_i > 0] \mathbb{P}[\sum_{b_j \in B \setminus S} D_j - c + \sum_{a_i \in N(S, \mathcal{A})} c_i < 0] \\ &= \mathbb{P}[\sum_{b_j \in S} (D_j - \mu_j) > \sum_{b_j \in B \setminus S} \mu_j + \sum_{a_i \in N(S, \mathcal{A})} c_i - \mu] \mathbb{P}[\sum_{b_j \in B \setminus S} (D_j - \mu_j) < c - \sum_{a_i \in N(S, \mathcal{A})} c_i - \sum_{b_j \in B \setminus S} \mu_j] \\ &\leq \mathbb{P}[\sum_{b_j \in S} (D_j - \mu_j) > \delta_{\mu}^{n-|S|}(\mathcal{A}) - \mu] \mathbb{P}[\sum_{b_j \in B \setminus S} (D_j - \mu_j) < -(\delta_{\mu}^{n-|S|}(\mathcal{A}) - c)] \\ &= (1 - \Phi(\frac{\delta_{\mu}^{n-|S|}(\mathcal{A}) - \mu}{\sqrt{|S|}\sigma})) (1 - \Phi(\frac{\delta_{\mu}^{n-|S|}(\mathcal{A}) - c}{\sqrt{n-|S|}\sigma})) \end{aligned}$$

Therefore, $\max_{S \subseteq B} \{\Pi(\mathcal{A}, S)\} \leq \max_{1 \leq k < n} (1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - \mu}{\sqrt{n-k}\sigma}))(1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - c}{\sqrt{k}\sigma}))$. Note that for any $1 \leq k < n$, there exists $S^k \subset B$, $|S^k| = k$ and $\delta_{\mu}^k(\mathcal{A}) = \sum_{a_i \in N(B \setminus S^k, \mathcal{A})} c_i + \sum_{b_j \in S^k} \mu_j$. Let $\bar{S}^k = B \setminus S^k$, and we have

$$\begin{aligned} \Pi(\mathcal{A}, \bar{S}^k) &= \mathbb{P}[\sum_{b_j \in \bar{S}^k} D_j - \sum_{a_i \in N(\bar{S}^k, \mathcal{A})} c_i > 0] \mathbb{P}[\sum_{b_j \in S^k} D_j - c + \sum_{a_i \in N(\bar{S}^k, \mathcal{A})} c_i < 0] \\ &= \mathbb{P}[\sum_{b_j \in \bar{S}^k} (D_j - \mu_j) > \sum_{a_i \in N(\bar{S}^k, \mathcal{A})} c_i + \sum_{b_j \in S^k} \mu_j - \mu] \mathbb{P}[\sum_{b_j \in S^k} (D_j - \mu_j) < c - \sum_{a_i \in N(\bar{S}^k, \mathcal{A})} c_i - \sum_{b_j \in S^k} \mu_j] \\ &= (1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - \mu}{\sqrt{n-k}\sigma}))(1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - c}{\sqrt{k}\sigma})) \end{aligned}$$

and thus, we also have $\max_{S \subseteq B} \{\Pi(\mathcal{A}, S)\} \geq \max_{1 \leq k < n} (1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - \mu}{\sqrt{n-k}\sigma}))(1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - c}{\sqrt{k}\sigma}))$ and the proof is complete. \square

Proposition 3 demonstrates that the PCIs not only have worst-case implications, but are also connected to the JG index, an index proposed under stochastic setting for average-case analysis. The result in Proposition 3 can be similarly extended to other demand distributions where $\{D_j - \mu_j\}_{1 \leq j \leq n}$ is IID. In that case, let $X_j = D_j - \mu_j$ for $1 \leq j \leq n$, and the JG index for a flexibility design \mathcal{A} can be expressed as

$$JG(\mathcal{A}) = \max_{1 \leq k < n} \mathbb{P}[\sum_{j=1}^{n-k} X_j > \delta_{\mu}^k(\mathcal{A}) - \mu] \cdot \mathbb{P}[\sum_{j=1}^k X_j < -(\delta_{\mu}^k(\mathcal{A}) - c)].$$

Proposition 3 also leads to an immediate corollary for comparing the JG indices of flexibility designs. Note that the design with the lower JG index is expected to be more effective than the design with the higher JG index.

COROLLARY 2. *Suppose \mathbf{D} is a random vector with mean μ , and $\{D_j - \mu_j\}_{1 \leq j \leq n}$ is IID, then for any two flexibility designs \mathcal{A}_1 and \mathcal{A}_2 , if $\delta_{\mu}^k(\mathcal{A}_1) \geq \delta_{\mu}^k(\mathcal{A}_2)$ for $1 \leq k \leq n-1$, then $JG(\mathcal{A}_1) \leq JG(\mathcal{A}_2)$.*

Because the JG index can be determined from the PCIs, it implies that the PCIs carry more information than the JG index. In the next subsection, we will present a flexibility index that incorporates this extra amount of information provided by the PCIs and demonstrate its effectiveness through numerical studies.

5.2. The JG-Sum Index and Computational Results

Motivated by the JG index and Proposition 3, we propose the JG-Sum (or JGS) index as follows.

DEFINITION 3. The JG^k index of a flexibility design \mathcal{A} , denoted by $JG^k(\mathcal{A})$, is defined as

$$JG^k(\mathcal{A}) = (1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - c}{\sqrt{k}\sigma}))(1 - \Phi(\frac{\delta_{\mu}^k(\mathcal{A}) - \mu}{\sqrt{n-k}\sigma})).$$

And the JGS index of \mathcal{A} is defined as

$$JGS(\mathcal{A}) := \sum_{k=1}^{n-1} JG^k(\mathcal{A}).$$

From the definition, it is clear that the JGS index can also be expressed as a function of $\{\delta_{\mu}^k(\mathcal{A}); 1 \leq k < n\}$. The quantity $JG^k(\mathcal{A})$ can be interpreted as the largest blocking probability achieved by a bottleneck that contains exactly k products. Note that JG^k is also closely related to the JG index, as $JG(\mathcal{A}) = \max_{1 \leq k < n} JG^k(\mathcal{A})$.

Intuitively, for two designs \mathcal{A}_1 and \mathcal{A}_2 , if $JG(\mathcal{A}_1)$ is significantly less than $JG(\mathcal{A}_2)$, then one would expect $JGS(\mathcal{A}_1)$ to be less than $JGS(\mathcal{A}_2)$. However, when $JG(\mathcal{A}_1) \approx JG(\mathcal{A}_2)$, then just comparing the blocking probability of the strongest bottleneck may not be enough to predict the better design. By contrast, the JGS index takes into account $n - 1$ bottlenecks, and therefore may serve as a better alternative.

In the rest of this subsection, we perform computational experiments to test the effectiveness of the PCIs, JG index and JGS index. In our computational tests, the samples of the product demands are generated from independent normal distributions, with demand for the i th product having mean μ_i , and standard deviation $\frac{1}{2}$. Because demand should never be negative, whenever a sample has a product with negative demand, we change the demand for that product to be zero. This modification would slightly change the actual mean and standard deviation, but the change would not have any significant effect on the numerical analysis. We always start with an initial design that is analogous to the dedicated design, where each product is produced by exactly one plant. We assume that in the initial design, the capacity in each plant is equal to the total expected demand for all the products the plant produces. That is, if \mathcal{A} is the initial design, then $\sum_{b_j \in N(a_i, \mathcal{A})} \mu_j = c_i, \forall 1 \leq i \leq m$. We then randomly generate 50 designs by adding 2 arcs randomly at every plant to the initial design.

In the first test, we have a balanced system with $m = n = 10$, $c_i = \mu_i = 1$ for $i = 1, 2, \dots, 10$. In the second test, we have an unbalanced system with $m = 7$, $n = 14$, $c_1 = c_2 = 3$, $c_3 = c_4 = c_5 = 2$, $c_6 = c_7 = 1$; and $\mu_i = 1$ for $i = 1, 2, \dots, 14$. In the third test, we again have $m = 7$, $n = 14$, with μ_i being chosen uniformly randomly from 0.5 to 1.5, and $c_1 = \sum_{i=1}^3 \mu_i$, $c_2 = \sum_{i=4}^6 \mu_i$, $c_3 = \sum_{i=7}^8 \mu_i$, $c_4 = \sum_{i=9}^{10} \mu_i$, $c_5 = \sum_{i=11}^{12} \mu_i$, $c_6 = \mu_{13}$, $c_7 = \mu_{14}$.² In the first test, the initial design is simply the dedicated design with $n = 10$, while the initial design for the second and the third test is illustrated in Figure 2.

To test the accuracy of flexibility indices, we perform the following procedure. We first generate $500n$ demand instances, \mathbf{d}^i , $i = 1, \dots, 500n$, where each \mathbf{d}^i is drawn from the demand distribution previously described. For each pair of flexibility designs \mathcal{A}_1 and \mathcal{A}_2 , we consider

$$M = \frac{\sum_{i=1}^{500n} (P(\mathbf{d}^i, \mathcal{A}_1) - P(\mathbf{d}^i, \mathcal{A}_2))}{500n}$$

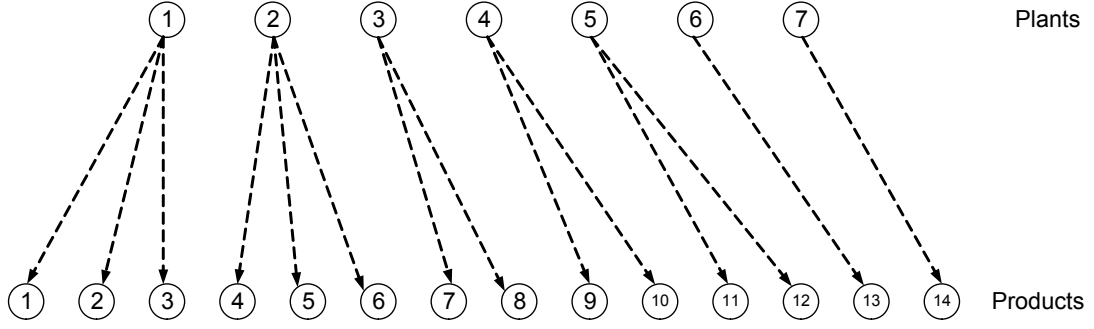


Figure 2 Initial Design for Test 2 and 3

$$\text{and } SE = \sqrt{\frac{\sum_{i=1}^{500n} (P(\mathbf{d}^i, \mathcal{A}_1) - P(\mathbf{d}^i, \mathcal{A}_2) - M)^2}{(500n - 1)(500n)}}$$

where M and SE are the sampled mean and standard error of the mean of $P(\mathbf{d}^i, \mathcal{A}_1) - P(\mathbf{d}^i, \mathcal{A}_2)$. If $M > 2 \cdot SE$ (or $M < -2 \cdot SE$), then we have statistically significant evidence that \mathcal{A}_1 (or \mathcal{A}_2) has higher expected sales. In this case, we would use \mathcal{A}_1 and \mathcal{A}_2 as one pair of designs to test the accuracy of our indices. We do not use \mathcal{A}_1 and \mathcal{A}_2 to test the accuracy of our indices if we do not have statistically significant evidence that identifies which design has higher expected sales. In the first, second, and the third tests, we had 682, 1124, and 1138 (out of $\binom{50}{2} = 1225$ possible) pairs of designs with statistically significant evidence that one design has higher expected sales than the other.

For each pair of \mathcal{A}_1 and \mathcal{A}_2 where we test the accuracy of the indices, we say that the index is “correct” about \mathcal{A}_1 and \mathcal{A}_2 if it correctly identifies the design with the higher expected sales; we say the index is “incorrect” about \mathcal{A}_1 and \mathcal{A}_2 if it incorrectly predicts the design with the higher expected sales, and finally, we say the index is “indecisive” between \mathcal{A}_1 and \mathcal{A}_2 , if it does not suggest which design is better.

The results of our computational tests are presented in Table 1. T1, T2 and T3 represent the three test settings. The column under “Indices” represents the flexibility indices in our test; the columns “Correct”, “Incorrect”, and “Indecisive” represent the number of the design pairs where the flexibility index is correct, incorrect, and indecisive, respectively; and finally the columns “Correct %”, “Incorrect %”, and “Indecisive %” represent the percentages of the instances where the index is correct, incorrect, and indecisive.

For the indices in our computational test, “PCI” compares all of the PCIs of \mathcal{A}_1 and \mathcal{A}_2 , and predicts a winner if and only if one of \mathcal{A}_1 and \mathcal{A}_2 is *strictly* more robust than another; “PCI-Sum” compares the sum of all the PCIs of \mathcal{A}_1 and \mathcal{A}_2 , while “JG” and “JG-Sum” compare the JG and JGS indices of \mathcal{A}_1 and \mathcal{A}_2 previously described. Under “PCI-Sum”, “JG” and “JG-Sum”, the index is indecisive if and only if the index of \mathcal{A}_1 is equal to the index of \mathcal{A}_2 .

	Indices	Correct	Incorrect	Indecisive	Correct %	Incorrect %	Indecisive %
T1	PCI	553	0	129	81.09%	0%	18.91%
	PCI-Sum	580	0	102	85.04%	0%	14.96%
	JG	520	5	157	76.25%	0.73%	23.02%
	JG-Sum	599	1	82	87.83%	0.15%	12.02%
T2	PCI	517	0	607	46.00%	0%	54.00%
	PCI-Sum	937	71	116	83.36%	6.32%	10.32%
	JG	840	3	281	74.73%	0.27%	25.00%
	JG-Sum	1108	14	2	98.58%	1.25%	0.18%
T3	PCI	63	0	1075	5.54%	0%	94.46%
	PCI-Sum	978	160	0	85.94%	14.06%	0 %
	JG	850	70	218	74.69%	6.15%	19.60%
	JG-Sum	1105	33	0	97.10%	2.90%	0 %

Table 1 Prediction of Different Flexibility Indices

We note a few important observations from the computational test. First, Table 1 indicates that while the PCI never incorrectly predicts the better design, it is too indecisive, especially in T3, where we have an unbalanced, non-IID demand system. The PCI-Sum is a good alternative when the PCI is indecisive, but it does incorrectly predict the better design in a significant portion of the instances.

Second, while the accuracy of the JG index is very good, like the PCI, it also tends to be indecisive at times. Interestingly, the JGS index is rarely indecisive, yet it rarely makes a mistake. In fact, the JGS predicts more winners than any other indices, yet its error percentages (shown as Incorrect %) are always comparable to that of JG, and much lower than that of PCI-Sum. This observation outlines the advantage of looking at *all of the bottlenecks generated by the PCIs*, instead of just one bottleneck.

Notice that

$$JGS(\mathcal{A}) = \sum_{i=1}^{n-1} f^k(\delta_{\mu}^k(\mathcal{A})), \text{ where } f^k(t) = (1 - \Phi(\frac{t-c}{\sqrt{k}\sigma}))(1 - \Phi(\frac{t-\mu}{\sqrt{n-k}\sigma})).$$

Thus, $JGS(\mathcal{A})$ can be thought of as a non-linear weighted sum of the PCIs, where the weight functions, f^k for $1 \leq k < n$, are motivated by the intuition of the JG index. To get a better understanding of why JGS outperforms PCI-Sum, consider a case where the values of $\delta_{\mu}^k(\mathcal{A})$ are much larger than c or μ for some $1 \leq k < n$. Intuitively, this implies that the bottlenecks containing k products at \mathcal{A} are not tight, and thus should have little effects on the expected sales. The JGS index correctly integrates this intuition, as $f^k(t)$ decreases exponentially quickly with t . However, for the PCI-Sum index, it does nothing to incorporate the fact that the bottlenecks containing k products at \mathcal{A} are not tight.

We also perform two additional numerical studies (Test 4 and Test 5) to investigate the robustness of the JG and JGS indices when the demands have different standard deviations. In this case, the

JG and JGS indices are computed using a *constant and incorrect* standard deviation. For both numerical studies, we set the number of plants and products, plant capacities, and the means of product demands to be the same as those in Test 3. In Test 4, we let $\sqrt{\mathbb{E}[(D_j - \mu_j)^2]} = \frac{1}{2}\mu_j$ for all $1 \leq j \leq 14$; in Test 5, we let $\sqrt{\mathbb{E}[(D_j - \mu_j)^2]} = R_j\mu_j$ for all $1 \leq j \leq 14$, where R_i is randomly generated from 0.2 to 0.6. The JG and JGS indices are computed assuming that the standard deviations of all demands are $\frac{1}{2}$, which is significantly different from the actual standard deviations.

Our tests suggest that the JGS is still a fairly accurate index when (i) the demands have different standard deviations and (ii) when the standard deviation used for computing the index is significantly off from the true value. The detailed computational results of Test 4 and Test 5 are presented in Table 2.

Finally, we briefly comment on applying the JG and JGS indices when the demand does not have normal distribution and $\{X_j = D_j - \mu_j\}_{1 \leq j \leq n}$ is IID. In this case, one can still define the JG and JGS index as

$$JG(\mathcal{A}) := \max_{1 \leq k < n} JG^k(\mathcal{A}), \quad JGS(\mathcal{A}) := \sum_{k=1}^{n-1} JG^k(\mathcal{A}),$$

$$\text{where } JG^k(\mathcal{A}) = \mathbb{P}\left[\sum_{j=1}^{n-k} X_j > \delta_{\mu}^k(\mathcal{A}) - \mu\right] \cdot \mathbb{P}\left[\sum_{j=1}^k X_j < -(\delta_{\mu}^k(\mathcal{A}) - c)\right].$$

Of course, the exact value of $JG^k(\mathcal{A})$ may be difficult to compute, and when that is the case, one may consider the approximate JG and JGS indices by assuming $\{X_j\}_{1 \leq j \leq n}$ is normal. Intuitively, the approximated JGS should work reasonably well especially compared to the PCI-Sum index, as the approximated JGS puts less weights on the bottlenecks that are not tight. Moreover, we expect the approximated JGS index to perform especially well if the tail of each X_j is reasonably similar to that of the normal distribution.

	Indices	Correct	Incorrect	Indecisive	Correct %	Incorrect %	Indecisive %
T4	PCI	63	0	1075	5.54%	0%	94.46%
	PCI-Sum	981	157	0	86.20%	13.80%	0 %
	JG	831	84	223	73.02%	7.38%	19.60%
	JG-Sum	1085	53	0	95.34%	4.66%	0 %
T5	PCI	63	0	1064	5.59%	0%	94.41%
	PCI-Sum	954	173	0	84.65%	15.35%	0 %
	JG	821	92	214	72.85%	8.16%	18.99%
	JG-Sum	1140	87	0	92.28%	7.72%	0 %

Table 2 Robustness of Incorrect Standard Deviation

6. Generating Effective Flexibility Design

In this section, we propose a class of heuristics for finding effective flexibility designs. The heuristics we propose start with a flexibility design \mathcal{A} , and iteratively add arcs to improve \mathcal{A} . First, we describe the general framework of a single iteration in our heuristics.

We compute $\delta_{\mu}^k(\mathcal{A})$ for all $1 \leq k \leq n-1$ and identify sets $S^k \subsetneq B$, where $\delta_{\mu}^k(\mathcal{A}) = \sum_{b_j \in S^k} \mu_j + \sum_{a_i \in N(B \setminus S^k, \mathcal{A})} c_i$. As discussed earlier, S^k and $N(B \setminus S^k, \mathcal{A})$ correspond to the tightest bottleneck of \mathcal{A} containing k products. Then for each $1 \leq k < n$, we select weight function $f^k(\cdot)$, and for the tightest bottleneck of \mathcal{A} containing k products, we assign the bottleneck with a weight of $f^k(\delta_{\mu}^k(\mathcal{A}))$. In the heuristic, we add the next arc that relaxes a subset of the bottlenecks in

$$\{S^k, N(B \setminus S^k, \mathcal{A}); 1 \leq k < n\},$$

where the subset has the *largest possible total weight*. Note that to relax the bottleneck defined by S^k and $N(B \setminus S^k, \mathcal{A})$, one needs to add an arc (a_i, b_j) to \mathcal{A} , where $a_i \in A \setminus N(B \setminus S^k)$ and $b_j \in B \setminus S^k$.

The intuition of our heuristic is similar to an idea of Jordan and Graves (1995), where the authors propose adding an arc to relax the bottleneck corresponding to S^{k^*} , for some k^* such that $\Pi(\mathcal{A}, S^{k^*}) = JG(\mathcal{A})$. A challenge with this approach proposed by Jordan and Graves is that there are almost always multiple arcs that can relax the bottleneck corresponding to S^{k^*} , and it is unclear how to choose the best arc among them. By contrast, our heuristic considers simultaneously $n-1$ bottlenecks, and hence uses more information to choose the next arc.

Next, we formally describe this class of heuristics.

Algorithm 1 The Plant Cover Heuristics

- 1: Given: \mathcal{A} in a m plants n products system, and a budget of K arcs.
 - 2: Select a set of weight functions, $f^k(\cdot)$, for $1 \leq k \leq n-1$.
 - 3: **for** $t = 1, 2, \dots, K$ **do**
 - 4: Find $\delta_{\mu}^1(\mathcal{A})$, $\delta_{\mu}^2(\mathcal{A})$, ..., $\delta_{\mu}^{n-1}(\mathcal{A})$, and their corresponding optimal solutions $(\mathbf{p}^1, \mathbf{q}^1)$, $(\mathbf{p}^2, \mathbf{q}^2)$, ..., $(\mathbf{p}^n, \mathbf{q}^n)$.
 - 5: Let $\Psi(x, y) = 1$ if $x = y = 0$ and $\Psi(x, y) = 0$ otherwise. For each $1 \leq i \leq m$, $1 \leq j \leq n$, compute $W(i, j) = \sum_{k=1}^{n-1} f^k(\delta_{\mu}^k(\mathcal{A})) \cdot \Psi(p_i^k, q_j^k)$.
 - 6: Find arc (a_{i^*}, b_{j^*}) such that $W(i^*, j^*) = \max\{W(i, j) | 1 \leq i \leq m, 1 \leq j \leq n\}$ (when there is a tie, we uniformly randomly select an arc with the maximum $W(i, j)$).
 - 7: Add (a_{i^*}, b_{j^*}) to \mathcal{A} .
 - 8: **end for**
 - 9: Return \mathcal{A} .
-

The plant cover heuristics described in Algorithm 1 defines a class of heuristics because one can choose different sets of weight functions $f^k(\cdot)$. One natural set of weight functions is $f^k(t) = 1$ for any $t \in \mathbb{R}$ and $1 \leq k < n$. For this set of weight functions, the heuristic will always add the arc that can relax the largest number of bottlenecks defined by S^k for $1 \leq k < n$. Another natural set of weight functions is $f^k(t) = (1 - \Phi(\frac{t-c}{\sqrt{k}\sigma}))(1 - \Phi(\frac{t-\mu}{\sqrt{n-k}\sigma}))$, and in this case, the weights are the probabilities that the bottleneck corresponding to S^k blocks \mathcal{A} from achieving the same sales as full flexibility. To avoid round-off errors factoring into our algorithm, we make a small modification on this set of weight functions by letting

$$f^k(t) = (1 - \Phi(\frac{t-c}{\sqrt{k}\sigma}))(1 - \Phi(\frac{t-\mu}{\sqrt{n-k}\sigma})) + 10^{-4}, \forall 1 \leq k < n$$

in our computational experiments. This modification also has the advantage of guaranteeing some weights to all of S^k , which may improve the robustness of the algorithm when σ is significantly off from the true standard deviations.

We note that another similar heuristic proposed in the literature is the expander heuristic proposed by Chou et al. (2011). In particular, given a design \mathcal{A} , the expander heuristic finds bottlenecks $A \setminus a_{i^*}$ and $N(a_{i^*}, \mathcal{A})$ and $B \setminus b_{j^*}$ and $N(b_{j^*}, \mathcal{A})$, where

$$i^* = \arg \min_{1 \leq i \leq m} \frac{\sum_{b_j \in N(a_i, \mathcal{A})} \mu_j}{C_i}, j^* = \arg \min_{1 \leq j \leq n} \frac{\sum_{a_i \in N(b_j, \mathcal{A})} C_i}{\mu_j},$$

and adds arc (a_{i^*}, b_{j^*}) to \mathcal{A} . Note that $(A \setminus a_{i^*}, N(a_{i^*}, \mathcal{A}))$ is the tightest bottleneck containing $m - 1$ plants, $(B \setminus b_{j^*}, N(b_{j^*}, \mathcal{A}))$ is the tightest bottleneck containing $n - 1$ products, and (a_{i^*}, b_{j^*}) is the arc that relaxes both bottlenecks. In the interest of space, we leave out the details of the heuristic when (a_{i^*}, b_{j^*}) is already in \mathcal{A} , and refer the interested readers to Algorithm 1 presented in Chou et al. (2011).

Table 3 presents numerical results comparing the designs generated by the plant cover heuristics and other heuristics in the literature. Other than the plant cover heuristics, the heuristics presented in Table 3 include (i) the design with the highest expected sales among 50 randomly generated designs; (ii) incomplete 3-chain, which attempts to construct a 3-chain design described in Hopp et al. (2004) using K available arcs (see Figure 4a and 4b); and (iii) the design generated by the expander heuristic in Chou et al. (2011). Finally, the performance of the full flexibility design is also computed as a reference.

Similar to Tests 1, 2, and 3 in Section 5.2, we consider three sets of tests, where we have $m = n = 10$ in Test 1 and $m = 7, n = 14$ in Tests 2 and 3. The capacities of the plants and the demand distribution of the products are chosen exactly as in Tests 1, 2, and 3 in Section 5.2. Also, we start with an initial design that is analogous to the dedicated design, where each product is produced

by exactly one plant, and the capacity of each plant is equal to the total expected demand of all of its product. We add 15 arcs in Test 1 and 10 arcs in Tests 2 and 3.

In Table 3, we present the average sales of each design produced by different heuristics, under $500n$ randomly generated demand instances. UW-PCI represents Algorithm 1 with $f^k(t) = 1$ for $1 \leq k \leq n-1$ (think of UW-PCI as the heuristic that has a uniform weight on each of the minimal bottlenecks containing $k = 1, 2, \dots, n$ products), and W-PCI represents Algorithm 1 with $f^k(t) = (1 - \Phi(\frac{t-c}{\sqrt{k}\sigma}))(1 - \Phi(\frac{t-\mu}{\sqrt{n-k}\sigma}))$ for $1 \leq k \leq n-1$ (think of W-PCI as the heuristic that has different weight on each of the each of the minimal bottlenecks containing $k = 1, 2, \dots, n$ products).

One possible risk measure when studying the robustness of a design \mathcal{A} under stochastic demand is the threshold value x such that the likelihood that the sales is lower than x is equal to $p\%$. This metric is also known as the p -th percentile of the sales of \mathcal{A} . To compare the different designs using different percentiles, we plot in Figure 3 the empirical CDF for each of these designs. The empirical CDFs of UW-PCI and W-PCI are highlighted with dashed lines.

Figure 3 suggests, that under each of the three tests, the empirical distribution of the sales of full flexibility (stochastically) dominates every other design as expected. More interestingly, the distributions of the sales of the designs generated from UW-PCI and W-PCI almost (or completely) coincide with each other, and these two PCI-related designs dominate the designs generated by all other heuristics under the three tests. This implies that the designs generated by UW-PCI and W-PCI perform better not only in expected sales, but also in *every empirical percentile*. In particular, the 25th percentile of all designs are presented in Table 3. Finally, Table 3 also presents the worst ratio of the sales of a design to that of full flexibility under all demand instances.

The PCI concept motivates another heuristic in which we add the arc that makes the biggest improvement in the PCIs. More specifically, given a flexibility design \mathcal{A} , the heuristic searches over all possible arcs and adds the arc (a_{i^*}, b_{j^*}) that provides the biggest increase in

$$\sum_{k=1}^{n-1} f^k(\delta_{\mu}^k(\mathcal{A} \cup \{(a_{i^*}, b_{j^*})\})) - \sum_{k=1}^{n-1} f^k(\delta_{\mu}^k(\mathcal{A})),$$

where $f_k(\cdot)$ is defined either as the UW-PCI or the W-PCI. It turns out that this heuristic does not perform as well as either UW-PCI or the W-PCI. The reason is that at some iterations of the heuristic, we have design \mathcal{A} , where there does not exist an arc that improves any of its PCIs, for example, $\delta_{\mu}^k(\mathcal{A}) = \delta_{\mu}^k(\mathcal{A} \cup \{(a_i, b_j)\})$ for any $1 \leq k < n$ and any arc (a_i, b_j) . In those instances, the heuristic end up adding a random arc, which degrades its performance. This heuristic is also much slower than Algorithm 1, as it has to compute $\sum_{k=1}^{n-1} f^k(\delta_{\mu}^k(\mathcal{A} \cup \{(a_i, b_j)\}))$ for every arc $(a_i, b_j) \notin \mathcal{A}$ at each iteration.

While the expander heuristic performed worse than other heuristics under IID demand (test settings 1 and 2), it performed better under asymmetric demand (test setting 3). More importantly,

System	Heuristic	Avg. Sales	25th Pct.	Worst Ratio
Test Setting 1, add 15 arcs	UW-PCI	9.36	8.95	84.8%
	W-PCI	9.36	8.98	85.5%
	Random	9.26	8.77	80.8%
	3-Chain	9.30	8.84	81.1%
	Expander	8.88	8.35	72.7%
	Full Flexibility	9.41	9.01	-
Test Setting 2, add 10 arcs	UW-PCI	13.22	12.70	83.7%
	W-PCI	13.23	12.73	87.5%
	Random	13.09	12.58	79.8%
	3-Chain	13.17	12.61	81.9%
	Expander	13.07	12.40	76.0%
	Full Flexibility	13.29	12.79	-
Test Setting 3, add 10 arcs	UW-PCI	11.63	11.19	85.5%
	W-PCI	11.63	11.19	85.5%
	Random	11.50	11.04	81.8%
	3-Chain	11.54	11.00	79.4%
	Expander	11.54	10.96	78.0%
	Full Flexibility	11.70	11.27	-

Table 3 Comparison between the plant cover heuristics and others heuristics

the expander heuristic is very flexible and similar to the general framework provided by Algorithm 1. For these reasons, we plot in Figure 5 the performances of UW-PCI and the expander heuristic as a function of the number of arcs added. The performances are measured by the expected sales of the designs generated by both heuristics (normalized by the expected sales of full flexibility). The plot of W-PCI is omitted because while it is slightly better than UW-PCI numerically, one cannot really differentiate W-PCI from UW-PCI on the plot.

The computational results in Figure 5 again demonstrate the effectiveness of the plant cover heuristics. Moreover, because the expander heuristic only uses two bottlenecks in adding arcs to the existing flexibility design, it demonstrates the advantage of looking at all of the bottlenecks generated by the PCIs.

Note that the W-PCI heuristic attempts to maximize the JGS index while the UW-PCI heuristic attempts to maximize the PCI-Sum during every iteration. Because the JGS index significantly outperformed the PCI-Sum index in the numerical studies in Section 5.2, it may seem surprising that the W-PCI heuristic only outperforms UW-PCI by an extremely small margin. We think there are two reasons for this. First, because both W-PCI and UW-PCI heuristics add one arc at each iteration, both heuristics often end up adding the same arc, although they seek to maximize different objectives. Second, in the numerical tests, the designs identified by the UW-PCI heuristic are already very close to optimal, and therefore, the W-PCI heuristic has very little room to make further improvements.

We also present our computational studies when product demands are correlated to verify the robustness of Algorithm 1. Intuitively, because Theorem 3 holds for any uncertainty set with

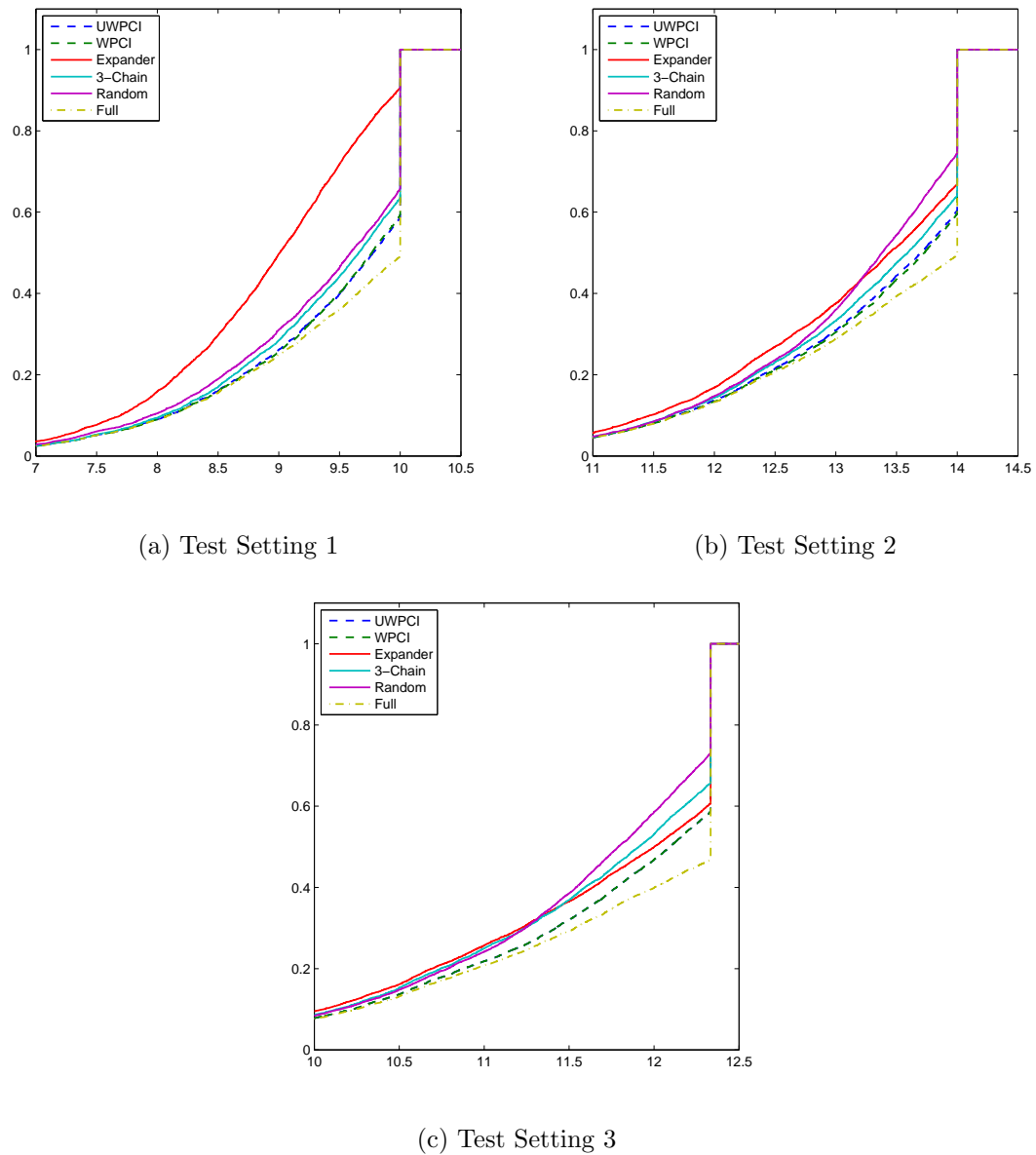


Figure 3 Empirical Cumulative Distribution Functions (CDF) of the Sales

symmetric uncertainties, the plant cover heuristics should still perform favorably compared to heuristics such as the expander heuristic when the product demand correlations are symmetric. This is illustrated in Figure 6, where the product demands have the same marginal distributions as test setting 3, while having the same pairwise positive and negative correlations. The correlation coefficients were chosen to be ± 0.03 . Again, we plot the ratios between the expected sales of the designs generated by both heuristics to that of full flexibility. Note that the performances of both heuristics are slightly lower under the negative correlation setting. This is because the full flexibility

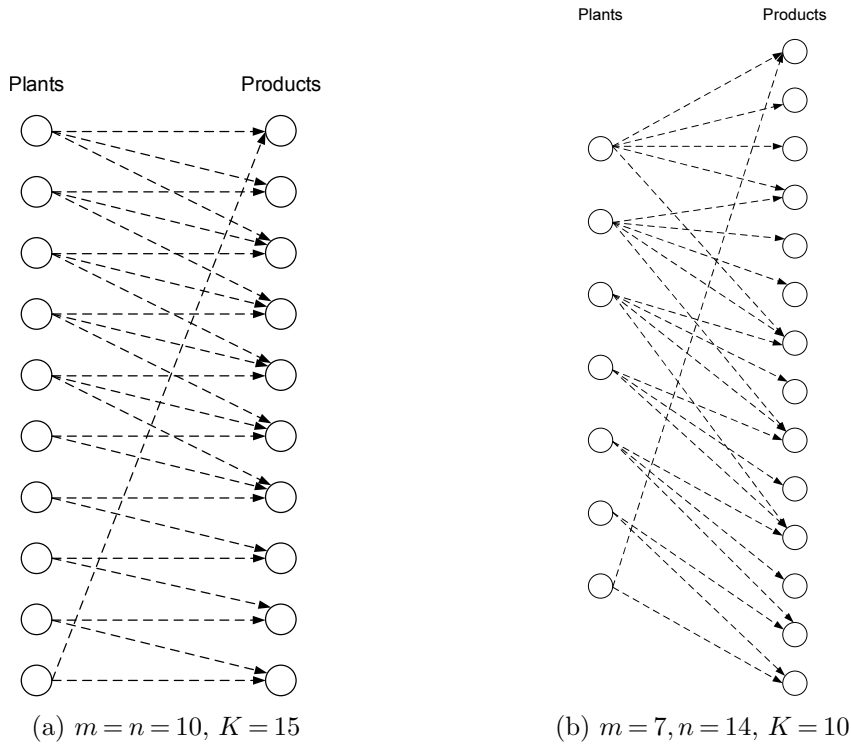


Figure 4 Incomplete 3-Chains

design does extremely well under the negative correlation setting, which makes the ratio of the expected sales of the designs generated by heuristics to that of full flexibility smaller.

A limitation of the plant cover heuristics is that when systems have strong positive or negative correlations in a small subset of products, there are no straightforward methods to take the correlations into account. However, strong correlations in a small subset of products typically make the problem of finding effective designs easy, as one can resort to intuitions such as “chaining” together all the products with strong negative correlation (see Jordan and Graves (1995)). Indeed, the most difficult instances in designing flexibility are when there is no correlation, or the correlations are equally spread across products. Those are the instances where the plant cover heuristics excel.

7. Discussion and Conclusion

The objective of this paper is to provide insights from analyzing the worst-case performances of process flexibility designs. For this purpose, we first introduce the plant cover indices (PCIs), where the plant cover index (PCI) at k can be thought of as the tightest bottleneck containing k products. We prove that a general class of worst-case performance measures can be expressed as functions of the PCIs. This immediately leads to two important observations under this general class of worst-case performance measures: first, the set of all PCIs is a sufficient statistic for computing the worst-case performances of any flexibility design, and second, the PCIs induce a partial ordering

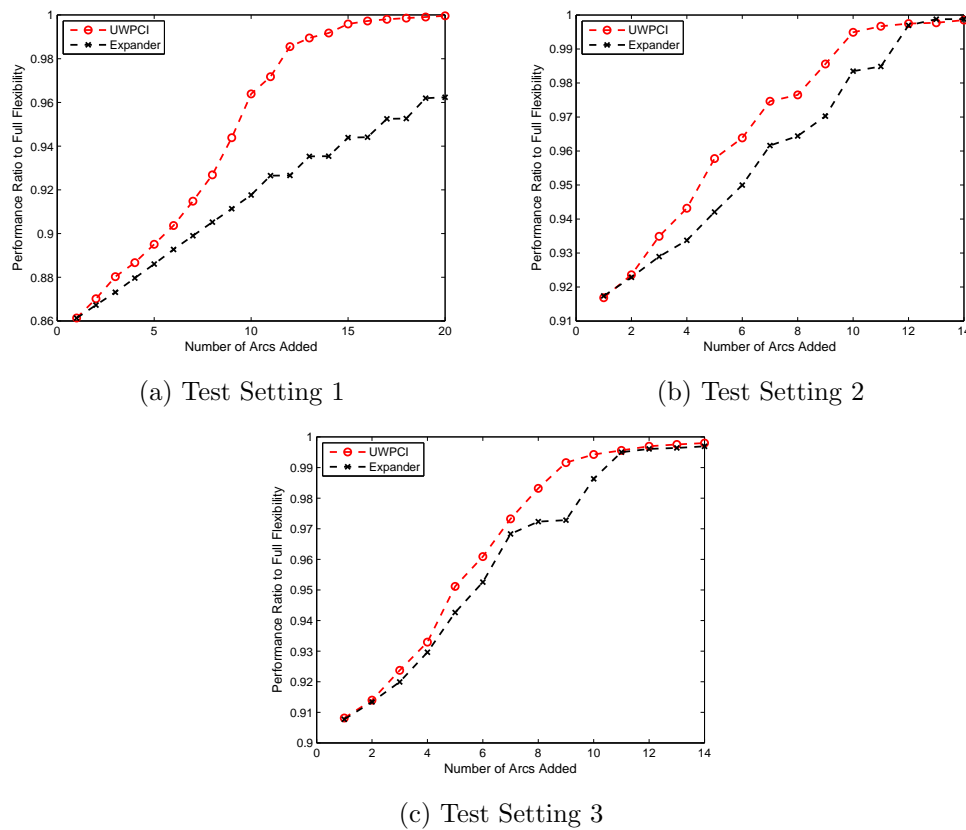


Figure 5 Plant Cover Heuristic vs Expander Heuristic

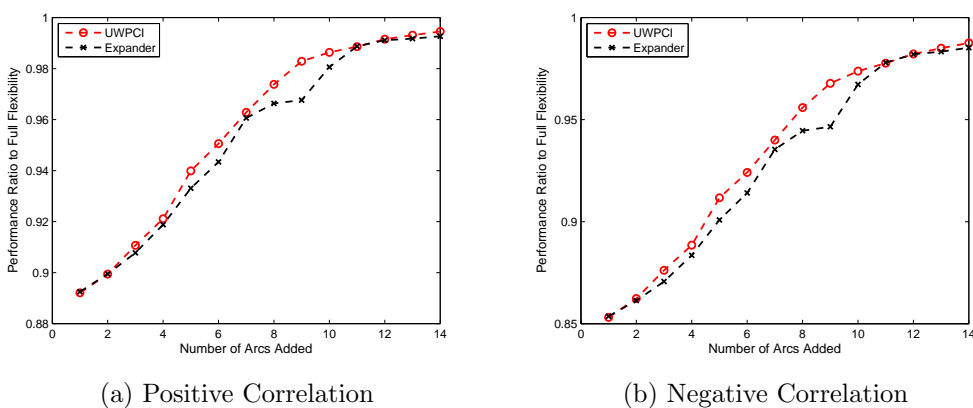


Figure 6 Plant Cover Heuristic vs Expander Heuristic under Demand Correlations

for comparing flexibility designs. The second observation is then applied to prove that under a balanced system with homogenous plants and products, the long chain flexibility design has a better worst-case performance than any design where the degree of each product node is two, or any connected design with $2n$ arcs. This result can be seen as a worst-case counterpart to an average-case result established in Simchi-Levi and Wei (2012).

Motivated by the theoretical results, we try to identify whether one can use the PCIs to quickly estimate and evaluate different designs from the average-case (i.e., expected sales) point of view. For this purpose, we prove that the classical Jordan and Graves (JG) index, developed to compare flexibility designs from the average-case point of view, can be determined as a function of the PCIs when product demands are independent and have the same variance. Furthermore, combining the JG index and PCIs, we propose a new index, the JGS index, which is significantly more accurate than the classical JG index under our numerical study. Finally, using the bottlenecks identified by the PCIs, we propose a class of (sequential) heuristics for generating flexibility designs. Computational study suggests that designs generated by our heuristics perform better than designs generated by other heuristics not only from the expected sales point of view, but also in terms of various risk measures.

Our study suggests several intriguing advantages of using the PCIs to evaluate and generate flexibility designs. First, it is very general. It can be applied to study unbalanced systems with unequal plant capacities, non-homogenous product demands, and even additional linear production constraints. Second, it requires little demand information, which is practically appealing in industries where demand is difficult to predict. In fact, the PCIs can be adapted to different levels of demand information; one can either study $\delta^k(\cdot)$, which does not use any product demand information, or study $\delta_{\mu}^k(\cdot)$, which uses only information about the expectations of product demands. Finally, the PCIs capture important characteristics of a flexibility design with just a few numerical values. In particular, for a design \mathcal{A} with n products, the PCIs correspond to $n + 1$ bottlenecks of \mathcal{A} , which is a significant reduction from the 2^n bottlenecks of \mathcal{A} . Despite this reduction, our result shows that the set of all PCIs is a sufficient statistic for determining the worst-case performances of \mathcal{A} .

It is appropriate to end the paper with some possible directions for future research. Theorem 4 shows that $\delta_{\mu}^k(\mathcal{A}_1) \geq \delta_{\mu}^k(\mathcal{A}_2)$ for all $1 \leq k < n$, if and only if $R(\mathcal{A}_1, U) \geq R(\mathcal{A}_2, U)$, for all $R \in \Gamma$ and any U that is symmetric around μ . An important open question is whether we can relax the condition $\delta_{\mu}^k(\mathcal{A}_1) \geq \delta_{\mu}^k(\mathcal{A}_2)$ for all $1 \leq k < n$ under a more restrictive class of worst-case performance measures. Moreover, numerical results from Section 5.2 suggest that the PCIs can be aggregated into a unique index, i.e., $\sum_{k=1}^n f^k(\delta_{\mu}^k(\cdot))$ for some set of weight functions $f^k(\cdot)$, that is very effective in simulation. It would be interesting if one can generate a unique index that either has novel theoretical guarantees or consistently outperforms the JGS index in numerical experiments.

Another direction is to develop methods for computing $\delta_{\mu}^k(\mathcal{A})$ for large systems. Our computational results suggest that when the number of plants or products is less than 100, off-the-shelf optimization solvers such as CPLEX can compute the PCIs within seconds. While this is sufficient in studying manufacturing systems, there are other networks (e.g., call centers and data networks) with a larger number of nodes. For example, in data networks, servers and tasks are analogous to

plants and products in a manufacturing system. Although researchers in computer science have designed exponential algorithms for computing the constraint bipartite vertex cover problems, i.e. $\delta^k(\mathcal{A})$, that work well in practice (see Fernau and Niedermeier (2001), Bai and Fernau (2008)), not much is known for computing $\delta_\mu^k(\mathcal{A})$. Therefore, if the PCIs are applicable to large size systems, it is important to identify methods for efficiently computing $\delta_\mu^k(\mathcal{A})$.

Acknowledgements. The authors would like to thank Mabel Chou, Stephen Graves, Sridhar Tayur, Chung-Piaw Teo, the associate editor of *Operations Research*, and the two anonymous referees of *Operations Research* for reading preliminary versions of this paper. Their comments and suggestions have greatly improved the quality of the paper. This research was supported by the National Science Foundation [contract CMMI-0758069], Masdar Institute of Science and Technology, the Ford-MIT Alliance, and the Natural Sciences and Engineering Research Council of Canada Postgraduate Scholarship. Part of this work was completed when the second author was a Ph.D. student at the MIT Operations Research Center.

Endnotes

¹Alternatively, Theorem 2 in this paper conditioned on $f(\cdot)$ being *monotonic* in sales under fixed total demand also holds under a slightly more general condition under symmetric demand uncertainty sets. In particular, the same results hold for any f such that for any $\mathbf{d}^1, \mathbf{d}^2 \in \Sigma(\mathbf{d})$, and for any designs $\mathcal{A}_1, \mathcal{A}_2$, we have $f(\mathbf{d}^1, \mathcal{A}_1) > f(\mathbf{d}^2, \mathcal{A}_2) \iff P(\mathbf{d}^1, \mathcal{A}_1) > P(\mathbf{d}^2, \mathcal{A}_2)$.

²The randomly generated $\boldsymbol{\mu}$ turned out to be $\mu_1 = 0.9170, \mu_2 = 1.2203, \mu_3 = 0.5001, \mu_4 = 0.8023, \mu_5 = 0.6468, \mu_6 = 0.5923, \mu_7 = 0.6863, \mu_8 = 0.8456, \mu_9 = 0.8968, \mu_{10} = 1.0388, \mu_{11} = 0.9192, \mu_{12} = 1.1852, \mu_{13} = 0.7045, \mu_{14} = 1.3781$.

References

- Aksin, O. Zeynep, Fikri Karaesmen. 2007. Characterizing the performance of process flexibility structures. *Operations Research Letters* **35**(4) 477–484.
- Bai, Guoqiang, Henning Fernau. 2008. Constraint bipartite vertex cover simpler exact algorithms and implementations. *Proceedings of the 2nd annual international workshop on Frontiers in Algorithmics*. FAW '08, Springer-Verlag, Berlin, Heidelberg, 67–78.
- Bertsimas, Dimitris, Robert Weismantel. 2008. *Optimization Over Integers*. Dynamic Ideas.
- Chou, Mabel, Chung-Piaw Teo, Huan Zheng. 2008. Process flexibility: design, evaluation, and applications. *Flexible Services and Manufacturing Journal* **20** 59–94.
- Chou, Mabel, Chung-Piaw Teo, Huan Zheng. 2011. Process flexibility revisited: The graph expander and its applications. *Operations Research* **59**(5) 1090–1105.
- Chou, Mabel C., Geoffrey A. Chua, Chung-Piaw Teo, Huan Zheng. 2010. Design for process flexibility: Efficiency of the long chain and sparse structure. *Operations Research* **58**(1) 43–58.
- Deng, Tianhu. 2013. Process flexibility design in unbalanced and asymmetric networks. Ph.D. thesis, University of California, Berkeley.

- Deng, Tianhu, Zuo-Jun Max Shen. 2013. Process flexibility design in unbalanced networks. *Manufacturing & Service Operations Management* **15**(1) 24–32.
- Fernau, Henning, Rolf Niedermeier. 2001. An efficient exact algorithm for constraint bipartite vertex cover. *Journal of Algorithms* **38**(2) 374 – 410.
- Graves, Stephen C., Brian T. Tomlin. 2003. Process flexibility in supply chains. *Management Science* **49** 907–919.
- Hopp, Wallace J., Eylem Tekin, Mark P. Van Oyen. 2004. Benefits of skill chaining in serial production lines with cross-trained workers. *Management Science* **50**(1) 83–98.
- Iravani, Seyed M., Mark P. Van Oyen, Katharine T. Sims. 2005. Structural flexibility: A new perspective on the design of manufacturing and service operations. *Management Science* **51**(2) 151–166.
- Iravani, Seyed MR, Bora Kolfal, Mark P Van Oyen. 2007. Call-center labor cross-training: It’s a small world after all. *Management Science* **53**(7) 1102–1112.
- Jordan, William C, Stephen C Graves. 1995. Principles on the benefits of manufacturing process flexibility. *Management Science* **41**(4) 577–594.
- Kuo, Sy-yen, W. Kent Fuchs. 1987. Efficient spare allocation for reconfigurable arrays. *Design & Test of Computers* **4**(1) 24–31.
- Simchi-Levi, David. 2010. *Operations Rules: Delivering Customer Value through Flexible Operations*. The MIT Press.
- Simchi-Levi, David, Yehua Wei. 2012. Understanding the performance of the long chain and sparse designs in process flexibility. *Operations research* **60**(5) 1125–1141.

Appendix

A. Proof for Proposition 2

We only provide the proof for $\delta^{n-k}(\mathcal{C}) \geq \delta^{n-k}(\mathcal{A})$ for $0 \leq k \leq \alpha\sqrt{n}$. It is easy to check that $\delta^k(\mathcal{C}) \geq \delta^k(\mathcal{A})$ for $0 \leq k \leq \alpha\sqrt{n}$ from symmetrical arguments.

First, we note that $\delta^{n-k}(\mathcal{C}) = k + 1$ for any $k < n$. Because by Remark 2, $\delta^{n-k}(\mathcal{A}) \leq |N(S, \mathcal{A})|$, it is sufficient to prove that for any $|\mathcal{A}| \leq 2n$, and for any integer $1 \leq k \leq \alpha\sqrt{n}$, there always exists some $S \subseteq B$, with $|S| = k$ and $|N(S, \mathcal{A})| \leq k + 1$.

Suppose there exists a counterexample \mathcal{A}^* in a balanced system of size n . That is, there exists some k , $1 \leq k \leq \alpha\sqrt{n}$, for which we cannot find $S \subseteq B$ with $|S| = k$ and $|N(S, \mathcal{A}^*)| \leq k + 1$. Without loss of generality, assume \mathcal{A}^* is such a design in the *smallest* balanced system (with the minimum n). Let $1 \leq k^* \leq \alpha\sqrt{n}$ be the integer for which we cannot find any $S \subseteq B$ with $|S| = k^*$ and $|N(S, \mathcal{A}^*)| \leq k^* + 1$. Also, let $B_1 = \{u|u \in B, |N(u, \mathcal{A}^*)| = 1\}$, $B_2 = \{u|u \in B, |N(u, \mathcal{A}^*)| = 2\}$ and $B_3 = \{u|u \in B, |N(u, \mathcal{A}^*)| \geq 3\}$.

Suppose we have some $u, u \in B_1$ with $(v, u) \in \mathcal{A}^*$ and $|N(v, \mathcal{A}^*)| \geq 2$. Let $\mathcal{A}' = \{(v', u') | (v', u') \in \mathcal{A}^*, u' \neq u, v' \neq v\}$. Then \mathcal{A}' is a design in a balanced system of size $n - 1$, and $|\mathcal{A}'| \leq 2n - 2$. By our assumption on the minimality of n , we can find some $S \subseteq B \setminus u$ such that $|S| = k^* - 1$ and $|N(S, \mathcal{A}')| \leq k^*$. But this implies that $|N(S \cup \{u\}, \mathcal{A}^*)| \leq k^* + 1$, and we have a contradiction. Thus, for any $u \in B_1$ with $(v, u) \in \mathcal{A}^*$ we have $N(v, \mathcal{A}^*) = 1$. That is, any plant v that is a neighbor of some $u \in B_1$ in \mathcal{A}^* has a degree one.

Suppose there exists $B_C \subset B_2$ such that all arcs incident to B_C form a single cycle. Then clearly, $|N(B_C, \mathcal{A}^*)| = |B_C|$. If $|B_C| \geq k^*$, then it is easy to check that we can find $S \subseteq B_C$ with $|S| = k^*$ and $|N(S, \mathcal{A}^*)| \leq k^* + 1$, which leads to a contradiction. If $|B_C| < k^*$, then let $\mathcal{A}' = \{(v', u') | (v', u') \in \mathcal{A}^*, u' \notin B_C, v' \notin N(B_C, \mathcal{A}^*)\}$. In this case, $|\mathcal{A}'| = |\mathcal{A}^*| - 2|B_C| \leq 2(n - |B_C|)$, and \mathcal{A}' is a flexibility design defined for a system with $n - |B_C|$ plants and $n - |B_C|$ products. By the minimality of n , we can find some $S \subseteq B \setminus |B_C|$ such that $|S| = k^* - |B_C|$ and $N(S, \mathcal{A}') \leq k^* - |B_C| + 1$. This implies that $N(S \cup B_C, \mathcal{A}^*) \leq k^* + 1$, which is again a contradiction. Hence, there is no $B_C \subset B_2$ such that all arcs incident to nodes B_C form a cycle.

Let G_2 be the bipartite graph with node sets $A_2 = N(B_2, \mathcal{A}^*)$, B_2 and arc set $\mathcal{A}_2 = \{(v', u') | (v', u') \in \mathcal{A}^*, u' \in B_2\}$. Because there does not exist any $B_C \subset B_2$ such that all arcs incident to nodes B_C form one cycle, G_2 contains no cycles. This implies that G_2 contains a number of components, T_1, T_2, \dots, T_z , and $|T_i \cap B_2| = |T_i \cap A_2| - 1$ for all $1 \leq i \leq z$. Because any v that is a neighbor of $u \in B_1$ is not in T_i for all $1 \leq i \leq z$, $z = \sum_{i=1}^z |A \cap T_i| - |B \cup T_i| \leq (n - |B_1|) - |B_2| \leq |B_3|$. Because $|\mathcal{A}^*| = 2n$, the average degree of nodes in B is 2. This implies that $|B_3| \leq |B_1|$, and therefore, $z \leq |B_1|$. Now, if $z \leq 2$, we have

$$z(k^* - |B_1|) + |B_1| + |B_3| \leq z(k^* - |B_1|) + 2|B_1| \leq k^* \leq n$$

and if $z > 2$, we have

$$\begin{aligned} z(k^* - |B_1|) + |B_1| + |B_3| &\leq z(k^* - |B_1|) + 2|B_1| \\ &\leq z(k^* - z + 2) \\ &\leq z(\alpha\sqrt{n} - z + 2) \\ &\leq \left(\frac{\alpha\sqrt{n}}{2} + 1\right)^2 \leq n. \end{aligned}$$

Hence, we always have $z(k^* - |B_1|) + |B_1| + |B_3| \leq n \implies (k^* - |B_1|) \leq \frac{|B_2|}{z}$. This implies that $\sum_{i=1}^z |T_i \cap A_2|/z$ is at least $k^* - |B_1|$, and hence, there exists $1 \leq i^* \leq z$ such that T_{i^*} has $k^* - |B_1|$ plant nodes. Therefore, we can find a set $S \subseteq T_{i^*} \cap B$ such that $|N(S, \mathcal{A}^*)| \leq k^* - |B_1| + 1$, which implies that $|N(S \cap B_1)| \leq k^* + 1$. This leads to a contradiction. Hence, we must have that for any $0 \leq k \leq \alpha\sqrt{n}$, we can find some $S \subseteq B$, $|S| = k$ with $\delta^{n-k}(\mathcal{A}^*) \leq |N(S, \mathcal{A}^*)| \leq k + 1$. \square

B. Counterexamples

B.1. Robustness and Expected Sales

This subsection presents an example to show that given two flexibility designs, the strictly more symmetrically robust design does not always have higher expected sales under IID demand. In this case, consider $n = m = 4$, $c_i = 1$ for $i = 1, 2, 3, 4$ and flexibility designs \mathcal{A}_1 and \mathcal{A}_2 in Figure 7. Because \mathcal{A}_1 is a long chain with 4 plants and 4 products, $\delta^k(\mathcal{A}_1) = 4 - k + 1$ for each $1 \leq k \leq 3$.

To find $\delta^k(\mathcal{A}_2)$ for $1 \leq k \leq 3$, we can either solve the three corresponding binary programs or enumerate over all possible subsets of B and apply Remark 2. Applying either method, we find $\delta^1(\mathcal{A}_2) = 4$, $\delta^3(\mathcal{A}_2) = 2$, and $\delta^2(\mathcal{A}_2) = 2$. In particular, $\delta^2(\mathcal{A}_2)$ is achieved by set $S = \{b_3, b_4\}$; that is,

$$\delta^2(\mathcal{A}_2) = \sum_{a_i \in N(B \setminus S, \mathcal{A}_2)} c_i = \sum_{a_i \in \{a_1, a_2\}} c_i = 2.$$

Thus, $\delta^k(\mathcal{A}_1) = \delta^k(\mathcal{A}_2)$ for $k = 0, 1, 3, 4$ and $\delta^2(\mathcal{A}_1) > \delta^2(\mathcal{A}_2)$, which implies that \mathcal{A}_1 is strictly more symmetrically robust than \mathcal{A}_2 . However, it is not always true that \mathcal{A}_1 has higher expected sales than \mathcal{A}_2 even under IID demand. In particular, when product demand is IID and the demand for each product is equal to 0 or 2 with an equal probability of 0.5, the expected sales of \mathcal{A}_1 is equal to 3, but the expected sales of \mathcal{A}_2 is equal to 3.125. The expected sales of \mathcal{A}_1 by \mathcal{A}_2 are computed by enumerating over all of the possible demand instances.

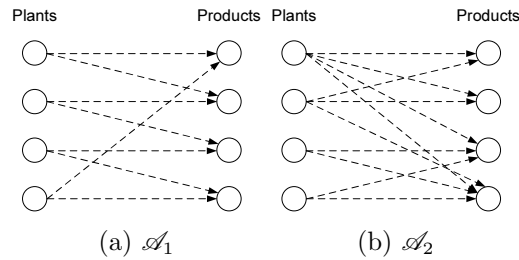


Figure 7 Designs \mathcal{A}_1 and \mathcal{A}_2

B.2. Robustness of the Long Chain and Designs with $2n$ Arcs

In Figure 8, we provide design \mathcal{A} with $n = 15$ nodes and 30 arcs, $c_i = 1$ for $1 \leq i \leq n$, where $\delta^k(\mathcal{A}) > \delta^k(\mathcal{C})$ for some $1 \leq k \leq n - 1$ (in this case, $k = 8$). To observe this, recall that $\delta^8(\mathcal{C}) = 15 - 8 + 1 = 8$. $\delta^8(\mathcal{A})$ is computed to equal to 9. While $\delta^8(\mathcal{A})$ can be computed by a case-by-case analysis, the easiest (and probably the most reliable) way to compute $\delta^8(\mathcal{A})$ is to use a computer program that either enumerates over all of the subset $S \subseteq B$ with $|S| = 8$ or solves the binary program corresponding to $\delta^8(\mathcal{A})$.

Because $\delta^8(\mathcal{A}) = 9 > \delta^8(\mathcal{C}) = 8$, Theorem 2 immediately implies that there exists some symmetric uncertainty set U where \mathcal{C} under the 15 plants and products system is strictly worse than \mathcal{A} under some worst-case measures.

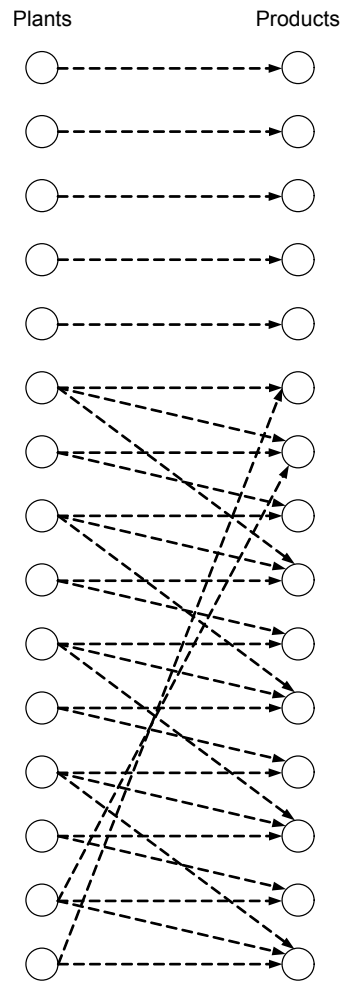


Figure 8 \mathcal{A} with 15 plants/products and 30 arcs

C. Additional Production Constraints

The concept of the PCIs introduced in Section 3 can be also extended to a model with an additional class of linear production constraints. In this case, $P(\mathbf{d}, \mathcal{A})$ is defined as the objective value of the following LP.

$$\begin{aligned}
 P(\mathbf{d}, \mathcal{A}) = \max \quad & \sum_{(a_i, b_j) \in \mathcal{A}} \theta_{ij} f_{ij} \\
 \text{s.t.} \quad & \sum_{a_i \in N(b_j, \mathcal{A})} f_{ij} \leq d_j, \forall b_j \in B
 \end{aligned}$$

$$\begin{aligned}
& \sum_{b_j \in N(a_i, \mathcal{A})} f_{ij} \leq c_i, \forall a_i \in A \\
& \sum_{(a_i, b_j) \in \mathcal{A}} \Phi_{hij} f_{ij} \leq \phi_h, \forall h = 1, 2, \dots, H \\
& 0 \leq f_{ij}, \forall (a_i, b_j) \in \mathcal{A} \\
& \mathbf{f} \in \mathbb{R}^{|\mathcal{A}|}.
\end{aligned}$$

For example, in some applications, an added flexibility arc (a_i, b_j) can be only utilized for p ($p < 1$) fraction of the capacity at plant i . In that case, we would have the additional constraint $f_{ij} \leq p \cdot c_i$.

Under this setting, we define the plant cover index, $\delta_{\mu}^k(\mathcal{A})$, for $0 \leq k \leq n$, as follows.

$$\begin{aligned}
\delta_{\mu}^k(\mathcal{A}) &= \min \sum_{i=1}^m c_i p_i + \sum_{j=1}^n \mu_j q_j + \sum_{h=1}^H \phi_h z_h \\
\text{s.t.} \quad & \sum_{j=1}^n q_j = k, \\
& p_i + q_j + \sum_{h=1}^H \Phi_{hij} z_h \geq 1, \forall (a_i, b_j) \in \mathcal{A} \\
& z_h \geq 0, \forall 1 \leq h \leq H, \\
& \mathbf{p} \in [0, 1]^m, \mathbf{q} \in \{0, 1\}^n, \mathbf{z} \in \mathbb{R}^h.
\end{aligned}$$

Note that the dual of the LP defining $P(\mathbf{d}, \mathcal{A})$ can be written as follows:

$$\begin{aligned}
P(\mathbf{d}, \mathcal{A}) &= \max \sum_{i=1}^m c_i p_i + \sum_{j=1}^n d_j q_j + \sum_{h=1}^H \phi_h z_h \\
\text{s.t.} \quad & p_i + q_j + \sum_{h=1}^H \Phi_{hij} z_h \geq 1, \forall (a_i, b_j) \in \mathcal{A} \\
& z_h \geq 0, \forall 1 \leq h \leq H, \\
& \mathbf{p} \in [0, 1]^m, \mathbf{q} \in [0, 1]^n, \mathbf{z} \in \mathbb{R}^h.
\end{aligned}$$

Consider the case where the dual problem has optimal integral solution(s). In this case, we can apply the same proof techniques as in Section 3, and develop the same result as Theorems 3 and 4 under this more general setting.

The dual problem has an optimal integral solution(s), when the system of inequalities,

$$\sum_{a_i \in N(b_j, \mathcal{A})} f_{ij} \leq d_j, \forall b_j \in B \tag{17}$$

$$\sum_{b_j \in N(a_i, \mathcal{A})} f_{ij} \leq c_i, \forall a_i \in A \tag{18}$$

$$\sum_{(a_i, b_j) \in \mathcal{A}} \Phi_{hij} f_{ij} \leq \phi_h, \forall h = 1, 2, \dots, H, \tag{19}$$

is *totally dual integral* (see Section 8.6 of Bertsimas and Weismantel (2008) for a more detailed discussion of this topic). For example, the set of inequalities in (17-19) is totally dual integral when all inequalities in (19) are of the form $f_{ij} \leq r_{ij}$.

Using the PCIs defined under this more general setting, we can define heuristics under the general framework outlined in Algorithm 1. When the dual of the LP defining $P(\mathbf{d}, \mathcal{A})$ has optimal integral solution(s), we expect the effectiveness of the heuristics under this more general setting to be comparable with the effectiveness of UW-PCI and W-PCI heuristics discussed in Section 6.