$\mathcal{I}^2\mathcal{SDS}$

The Institute for Integrating Statistics in Decision Sciences

Technical Report TR-2010-14

May 27, 2010

A Pattern Definition of the p-Efficiency Concept

Miguel A. Lejeune

Department of Decision Sciences

The George Washington University

A Pattern Definition of the p-Efficiency Concept

MIGUEL A. LEJEUNE*

George Washington University, Washington, DC, USA; `mlejeune@gwu.edu`

Abstract

This study revisits the celebrated p-efficiency concept introduced by Prékopa [23] and defines a p-efficient point (pLEP) as a combinatorial pattern. The new definition uses elements from the combinatorial pattern recognition field and is based on the combinatorial pattern framework for stochastic programming problems proposed in [16]. The approach is based on the binarization of the probability distribution, and the generation of a consistent partially defined Boolean function representing the combination (F, p) of the binarized probability distribution F and the enforced probability level p. A combinatorial pattern provides a compact representation of the defining characteristics of a pLEP and opens the door to new methods for the generation of pLEPs. We show that a combinatorial pattern representing a pLEP constitutes a strong and prime pattern and we derive it through the solution of an integer programming problem. Next, we demonstrate that the (finite) collection of pLEPs can be represented as a disjunctive normal form (DNF) and propose an mixed-integer programming formulation allowing for the construction of the DNF that is shown to be prime and irreducible. We illustrate the proposed method to a problem studied by Prékopa [25].

1 Introduction

The concept of p-efficiency, introduced by Prékopa [23], permits the solution of probabilistically constrained mathematical programming problems of the form

$$\min \quad g(x) \tag{1}$$

$$\text{subject to} \quad Ax \geq b \tag{2}$$

$$\mathbb{P}\left(h_j(x) \geq \xi_j,\ j \in J\right) \geq p \tag{3}$$

$$x \in \mathbb{R} \times \mathbb{Z} \quad . \tag{4}$$

The $|J|$-dimensional random vector ξ is discretely distributed and its components ξ_i are not required to be independent. We denote by x the m-dimensional vector of continuous ($\mathbb{R}$) and integer ($\mathbb{Z}$) decision variables, by p a prescribed reliability level, by $\mathbb{P}$ a probability measure, and by $g(x) : \mathbb{R}^m \rightarrow \mathbb{R}$ the objective function. The system of inequalities (2), with $A \in \mathcal{R}^{t \times m}$ and $b \in \mathbb{R}^t$, represents the set of deterministic constraints, and the *probabilistic* constraint (3) is a joint one that ensures that the $|J|$

*The author is supported by Grant # W911NF-09-1-0497 from the Army Research Office.

inequalities $h_j(x) \geq \xi_j$ ($j \in J$) hold jointly with a probability at least equal to p. The above programming problem is well-known to be non-convex and NP-hard, and has been receiving sustained attention [10, 15, 18, 19, 29]. Besides the p-efficiency concept, solution methods based on mixed-integer programming (MIP) approaches (e.g., [19, 29]) or robust optimization (e.g., [7]) have also been used to solve problem (1)-(4).

Within the p-efficiency approach, one has to elicit the p-efficient points of the probability distribution of ξ prior to reformulating (1)-(4) as an equivalent disjunctive or as an MIP problem.

Definition 1 *[23] Let $p \in [0, 1]$. A point $v \in \mathbb{R}^n$ is called a p-efficient point (pLEP) of the discrete probability distribution function F, if*

$$F(v) \geq p, \quad \textit{and} \tag{5}$$

$$\textit{there is no} \quad v' \leq v, v' \neq v \textit{ such that } F(v') \geq p\ . \tag{6}$$

A number of enumerative algorithms have been proposed [3, 15, 18, 25, 26] for the generation of pLEPs. Others elicit the pLEPs using a cone generation algorithm in [10], a primal-dual approach [9], or, most recently, through the solution of a mathematical programming problem [17].

In Section 2, we shall revisit the p-efficiency concept and define a pLEP as a pattern. The new definition is derived from the combinatorial pattern recognition [11, 28, 20, 33] framework for stochastic programming problems proposed in [16]. A combinatorial pattern provides a compact representation of the defining characteristics of a pLEP and opens the door to new mathematical approaches for the generation of pLEPs. In Section 3, we show that any combinatorial pattern representing a pLEP constitutes a *strong* [13] and *prime* [4] pattern and propose an integer programming formulation allowing for its generation. In Section 4, we represent the (finite) collection of pLEPs as a disjunctive normal form (DNF), propose an MIP formulation allowing for the construction of the DNF, and demonstrate show that the DNF is *prime* and *irreducible* [14]. Section 5 provides an illustration of the proposed approach to a problem studied by Prékopa [25]. Section 6 provides concluding remarks.

2 Combinatorial Pattern Modeling Framework - Redefining p-Efficiency

For self-containment purposes, we present in this section a comprehensive overview of the combinatorial pattern modeling framework developed to solve probabilistic programming problem. The reader is referred to [16] for more detailed explanations along with examples illustrated the approach. The pattern modeling framework involves the following steps: *(i)* the *binarization* of the probability distribution, and *(ii)* the representation of the combination (F, p) of the binarized probability distribution F and the enforced reliability level p as a *partially defined Boolean function* (pdBf). We shall see that this modeling framework permits to redefine a pLEP as a combinatorial pattern, which, in turn, allows for the use of a new approach to derive pLEPs.

Let Ω be the finite set of the possible *realizations* k of the $|J|$-dimensional random vector ξ with distribution function F. A realization k is represented by a $|J|$-dimensional deterministic vector ω^k.

Definition 2 *[16] Consider the Boolean parameter I^k. A realization k is called p-sufficient ($I^k = 1$) if and only if $\mathcal{P}(\xi \le \omega^k) = F(\omega^k) \ge p$ and is p-insufficient ($I^k = 0$) otherwise.*

Using the above concept, we partition Ω into two disjoint sets of p-sufficient Ω^+ and p-insufficient Ω^- realizations: $\Omega = \Omega^+ \bigcup \Omega^-$ with $\Omega^+ \bigcap \Omega^- = \emptyset$. Note that any p-efficient realization is p-sufficient, but that the converse is not necessarily true.

The binarization of the probability distribution consists in the mapping of each real-valued vector ω^k into a binary vector β^k. The value of its components β^k_{ij} is defined with respect to a set of *cut points* with the notation c_{ij} denoting the i^{th} *cut point* associated with component ξ_j.

$$\beta^k_{ij} = \begin{cases} 1 & \text{if } \omega^k_j \ge c_{ij} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

with

$$c_{i'j} < c_{ij} \ \Rightarrow\ \beta^k_{ij} \le \beta^k_{i'j} \quad \text{for any} \quad i' < i,\ j \in J,\ k \in \Omega\ . \tag{8}$$

This provides us with the n-dimensional binary vector $\beta^k = \left[\beta^k_{11}, \ldots, \beta^k_{n_1 1}, \ldots, \beta^k_{ij}, \ldots, \beta^k_{n_j j}, \ldots\right]$, which is a vertex of $\{0,1\}^n$, where $n = \sum_{j \in J} n_j$ is the sum of the number n_j of cut points for each component ξ_j.

The binarization approach defines the binary projection $\Omega_B = \Omega^+_B \bigcup \Omega^-_B$ of Ω, where Ω^+_B (resp., Ω^-_B) denotes the set of binarized p-sufficient (resp., p-insufficient) realizations, and provides a pdBf $g(\Omega^+_B, \Omega^-_B)$ defined by the pair of sets (Ω^+_B, Ω^-_B) such that $\Omega^+_B, \Omega^-_B \subseteq \{0,1\}^n$. The pdBf represents the combination (F, p) of a probability distribution F and a probability level p.

Evidently, the cut points are parameters whose values cannot be defined arbitrarily. Their values must be such that they preserve the disjointedness of the binary set projections Ω^+_B and Ω^-_B. Indeed, we want to prevent p-sufficient and p-insufficient realizations from having the same binary projection. This requires the use of a consistent [16] set of cut points and we generate the so-called sufficient-equivalent set [16] whose construction is immediate.

Definition 3 *[16] A sufficient-equivalent set of cut points C^e is consistent and includes a cut point c_{ij} for any value ω^k_j taken by any of the p-sufficient realizations on any component ξ_j:*

$$C^e = \{c_{ij} : c_{ij} = \omega^k_j, j \in J, k \in \Omega^+\}\ . \tag{9}$$

The pdBf $g(\Omega^+_B, \Omega^-_B)$ associated with the sufficient-equivalent set of cut points is referred to as the sufficient-equivalent pdBf.

In order to generate a pLEP, we cannot restrict our attention to the realizations included in the support Ω of the random variable ξ. Besides those, we also consider all points or realizations that can qualify as p-efficient. For k to be p-efficient, a necessary condition (see Definition 1) is that:

$$F_j(\omega^k_j) \ge p, j = 1, \ldots, |J|\ , \tag{10}$$

where F_j is the marginal probability distribution of ξ_j. Thus, for every j, we create the set of values Z_j

$$Z_j = \{\omega^k_j : F_j(\omega^k_j) \ge p, k \in \Omega,\ j = 1, \ldots, |J|\}\ , \tag{11}$$

define the direct product [25]

$$Z = Z_1 \times \ldots \times Z_j \times \ldots \times Z_{|J|} \ , \tag{12}$$

and obtain the extended set $\Omega \bigcup Z$ of realizations.

As shown in [16], the application of the binarization process to the extended set of realizations provides the representation of the upper (resp., lower) envelope of the integer hull of the p-insufficient (resp., p-sufficient) realizations. It permits the *elimination* of a number of points and the derivation of the set $\bar{\Omega}$ of *relevant* realizations. The eliminazion process rests on the following two principles. First, multiple realizations have the same binary image and we only keep one of them. Second, by definition of the sufficient-equivalent set of cut points, the necessary conditions (10) for p-efficiency can be rewritten as $\beta^k_{1j} = 1, j \in J$, and we can delete any realization k such that $\beta^k_{1j} = 0$ for any $j \in J$.

3 e^p-Patterns

3.1 Definition

The following concepts are used in the remaining part of the manuscript. The Boolean variables $\beta_{ij}, i = 1, \ldots, n_j, j \in J$ and their complements $\bar{\beta}_{ij}$ are called *literals*. A conjunction of literals $t = \bigwedge\limits_{ij \in P} \beta_{ij} \bigwedge\limits_{ij \in N} \bar{\beta}_{ij}$ with $P \bigcap N = \emptyset$ is a *term* [4, 14] with degree $d = |P| + |N|$ equal to the number of literals involved in the definition of t. The set P (resp., N) defines the set of non-complemented (resp., complemented) literals. A term t *covers* a realization k, which is denoted by $t(k) = 1$, if k satisfies all the conditions defined by t, or alternaltively stated, if the products of the values β^k_{ij} taken k on the literals β_{ij} defining t is equal to 1:

$$t(k) = 1 \quad \Leftrightarrow \quad \bigwedge_{ij \in P} \beta^k_{ij} \bigwedge_{ij \in N} \bar{\beta}^k_{ij} = 1 \tag{13}$$

$$\Leftrightarrow \quad \begin{cases} \omega^k_j \geq c_{ij}, \ ij \in P \\ \omega^k_j < c_{ij}, \ ij \in N \end{cases} . \tag{14}$$

The *coverage* of a term is the number of realizations covered by it. A *pattern* is also a conjunction of literals and can be viewed as a term that satisfies a "coverage condition". Indeed, a pattern is a subcube of the n-dimensional unit cube $\{0, 1\}^n$ that intersects $\bar{\Omega}^+_B$ (i.e., it covers at least one p-sufficient realization) but does not intersect $\bar{\Omega}^-_B$ (i.e., it does not cover any p-insufficient realization).

Using the above concepts, we redefine a p-efficient point as a combinatorial pattern, therefater referred to as a p-pattern and denoted by t^p.

Definition 4 *A term $t^p = \bigwedge\limits_{ij \in P} \beta_{ij}$ is called an e^p-pattern of the probability distribution function F at the reliability level $p \in [0, 1]$ if*

$$\bigvee_{k \in \bar{\Omega}^+_B} t^p(k) \geq 1 \ \textit{and} \ \bigwedge_{k \in \bar{\Omega}^-_B} t^p(k) = 0, \ \textit{and} \tag{15}$$

$$\textit{if for } t^{j'} = \beta_{i-1j'} \bigwedge_{ij \in P \setminus (ij')} \beta_{ij}, \ j' = 1, \ldots, |J|, \ \textit{we have} \bigvee_{k \in \bar{\Omega}^-_B} t^{j'}(k) \geq 1. \tag{16}$$

Definition 4 and Definition 1 enforce equivalent conditions. There is an immediate correspondence between (15) (resp., (16)) and condition (5) (resp., (6) in Definition 1.

3.2 Properties

An e^p-pattern has the following features. First, for the constraint (3) to hold, it is clear that at least one condition must be imposed with respect to each component of the random vector. Thus, an e^p-pattern includes at least one literal β_{ij} associated with each j and is of degree at least equal to $|J|$. Second, it was shown [16] that any pdBf $g(\bar{\Omega}_B^+, \bar{\Omega}_B^-)$ representing (F, p) is a positive monotone (isotone) Boolean function and it is known (see, e.g., [32]) that patterns representing an isotone function (i.e., an e^p-pattern for $g(\bar{\Omega}_B^+, \bar{\Omega}_B^-)$) do not need to contain complemented literals. Third, building on the result [16] that any prime pattern representing $g(\bar{\Omega}_B^+, \bar{\Omega}_B^-)$ includes exactly one literal for each j, it follows that any prime e^p-pattern is of degree $|J|$. A prime pattern is one that does not include any redundant literals [13]. Further, we have that:

Proposition 1 *An e^p-pattern is a prime pattern.*

Proof. Let $t^p = \bigwedge\limits_{ij \in P} \beta_{ij}$ be an arbitrary e^p-pattern. If t^p is not prime, then it includes one (or more) redundant literal and its removal does not transform t^p into a term that is not a pattern.
Consider an arbitrary literal $\beta_{ij'}, (ij') \in P$ and remove it from t^p, giving us the term $t' = \bigwedge\limits_{ij \in P \setminus (ij')} \beta_{ij}$. If $\beta_{ij'}$ is redundant, then t' should be an e^p-pattern, which requires that $\bigwedge\limits_{k \in \bar{\Omega}_B^-} t'(k) = 0$.
From (16), we have that t^p is an e^p-pattern if $t^{j'} = \beta_{i-1,j'} \bigwedge\limits_{ij \in P \setminus (ij')} \beta_{ij}$ is such that $\bigvee\limits_{k \in \bar{\Omega}_B^-} t^{j'}(k) \geq 1$. Compare $t^{j'} = \beta_{i-1,j'} \bigwedge\limits_{ij \in P \setminus (ij')} \beta_{ij}$ and $t' = \bigwedge\limits_{ij \in P \setminus (ij')} \beta_{ij}$ the term resulting from the removal of $\beta_{ij'}$. It is clear that any realization covered that by $t^{j'}$ is also covered by t'. Thus, $\bigvee\limits_{k \in \bar{\Omega}_B^-} t'(k) \geq \bigvee\limits_{k \in \bar{\Omega}_B^-} t^{j'}(k) \geq 1$. This shows that the term t' is not an e^p-pattern and thus that any e^p-pattern is prime. □
Also, since a pattern t is defined as strong [13] if there is no pattern t' such that the set of realizations covered by t' contains those covered by t, we have that:

Proposition 2 *A term is an e^p-pattern if and only if it is a p-strong pattern.*

Proof. Consider any e^p-pattern t^p with coverage $|Q^p|$. Only terms that impose less demanding requirements than those linked with t^p can cover a set of realizations that strictly include Q^p. The question to settle is whether there exists any such pattern t' ($\bigwedge\limits_{k \in \bar{\Omega}_B^-} t'(k) = 0$). If this is the case, t^p is not strong. Since each immediate "neighbor" $t^{j'}$ of t^p which requires the satisfaction of marginally weaker requirements covers at least one p-insufficient realization (16), the answer to the above question is negative.
Consider a strong pattern t with coverage $|Q|$. The definition of a strong p-pattern implies that: *(i)* t covers p-sufficient realizations exclusively and *(ii)* there is no other pattern whose set of covered realizations strictly includes Q. It is immediate to see that *(i)* implies (15). Similarly, *(ii)* implies that any term whose set of covered realizations strictly includes Q, and which is thus less demanding than t, is not a pattern. This is equivalent to (16). □

3.3 Generation

The proposed mechanism to generate e^p-patterns is based on the solution of a mathematical programming formulation. As shown in Section 3.2, a e^p-pattern has degree $|J|$, equal to the dimension of the random vector, and, while excellent enumerative methods [1, 2, 5, 12] have been proposed to derive combinatorial patterns, they are described [4, 31] as very computationally demanding to generate patterns of degree 4 or larger.

The derived p pattern t^p and the literals that it includes are defined with respect to the optimal values $(\mathbf{u}^*, \mathbf{w}^*)$ of the binary decision variables in the integer programming problem `IP1` [16]. We denote by u_{ij} (21) the binary decision variable indicating whether the corresponding literal β_{ij} is included ($u^*_{ij} = 1$) or not ($u^*_{ij} = 0$) in t^p. The binary variables w (22) are used to determine the coverage of t^p: $w^{k^*} = 1$ if the p-sufficient realization k is covered by t^p and is equal to 0 otherwise. The objective function (17) maximizes the coverage $|Q^p|$, $Q^p = \left\{k : w^{k^*} = 1, k \in \Omega^+_B\right\}$ of t^p. Since β^k_{ij} is a parameter indicating whether $\omega^k_j \geq c_{ij}$ ($\beta^k_{ij} = 1$) or not ($\beta^k_{ij} = 0$), it follows that (18) forces w^k to take value 0 if $k \in \Omega^+_B$ is not covered by t^p, while (19) precludes t^p from covering any $k \in \Omega^-_B$. Constraints (20) ensure that exactly one non-complemented literal per component ξ_j is included in t^p.

Theorem 1 *The optimal solution* $(\mathbf{u}^*, \mathbf{w}^*)$ *of* `IP1`

$$z = \max \sum_{k \in \bar{\Omega}^+_B} w^k \tag{17}$$

$$\textit{subject to} \quad \sum_{j \in J} \sum_{i=1}^{n_j} \beta^k_{ij} u_{ij} + |J|\,(1 - w^k) \geq |J|, \quad k \in \bar{\Omega}^+_B \tag{18}$$

$$\sum_{j \in J} \sum_{i=1}^{n_j} \beta^k_{ij} u_{ij} \leq |J| - 1, \quad k \in \bar{\Omega}^-_B \tag{19}$$

$$\sum_{i=1}^{n_j} u_{ij} = 1, \quad j \in J \tag{20}$$

$$u_{ij} \in \{0, 1\}, \quad j \in J, i = 1, \ldots, n_j \tag{21}$$

$$w^k \in \{0, 1\}, \quad k \in \bar{\Omega}^+_B \tag{22}$$

defines the e^p-pattern

$$t^p = \bigwedge_{\substack{\mathbf{u^*_{ij}}=1 \\ j \in J, i=1,\ldots,n_j}} \beta_{ij}$$

of degree $|J|$ with maximal coverage $|Q^p| = \left|\left\{k : \mathbf{w^{k^*}} = 1, k \in \bar{\Omega}^+_B\right\}\right|$.

Proof. The coverage constraints (18) and (19) ensure that t^p is a pattern, while (20) guarantee that the degree of t^p is equal to $|J|$. The optimal solution of `IP1` generates the pattern with largest coverage, and is thus strong and an e^p pattern (Proposition 2).

Denoting by $\mathbf{z}^*$ the optimal value of `IP1`, we obtain an upper bound $(|\Omega^+_B| - \mathbf{z}^*)$ on the number of e^p-patterns. It is important to note that the binarization process allows the removal (marginal quantile condition, same binary projection) of a very large number of p-insufficient realizations which facilitates the solution of `IP1`.

4 Disjunctive Normal Form of e^p-Patterns

In this section, we represent the collection of e^p-patterns as a disjunctive normal form (DNF) and propose an integrated and a sequential approaches to construct the DNF.

A disjunction $\bigvee_{s=1}^{S} t_s$ of terms t_s is called a DNF, and has degree d if $|P_s \bigcup N_s| \leq d, s = 1, \ldots, S$, with P_s (resp., N_s) denoting the set of uncomplemented (resp., complemented) literals involved in the definition of the term t_s. The objective is here to construct a DNF containing the exhaustive series of e^p-patterns t_s^p. Thus, the constructed DNF $f = \bigvee_{s=1}^{S} t_s^p$ is such that:

$$\begin{cases} f(k) \geq 1, \ k \in \Omega_B^+ \\ f(k) = 0, \ k \in \Omega_B^- \end{cases} . \tag{23}$$

4.1 Integrated Construction

We propose an MIP formulation allowing for the construction of a DNF (23) that contains the *minimal* number of patterns needed to cover all e^p- sufficient realizations. The method is *integrated* in that all the patterns included in the DNF are obtained through the solution of a single MIP problem. Every pattern included in the DNF is an e^p-pattern.

The following notations are used. Let $M = |\Omega_B^+| - \mathbf{z}^*$ be an upper bound (see Section 3.3) on the number of patterns needed to cover each $k \in \Omega_B^+$, with $\mathbf{z}^*$ denoting the optimal value of `IP1`. The continuous variable h_s (30) indicates whether the term t_s^p is included ($h_s^* = 1$) or not ($h_s^* = 0$) in the DNF (33) defined by the optimal solution of `IP1`. The binary variable w_s^k (31) indicates whether $k \in \Omega_B^+$ is covered ($w_s^{k^*} = 1$) or not ($w_s^{k^*} = 0$) by the pattern t_s^p included ($h_s^* = 1$) in the DNF. The binary variable $u_{ij,s}$ (32) indicates whether β_{ij}^k is included ($u_{ij,s}^* = 1$) or not ($u_{ij,s}^* = 0$) in the definition of t_s^p. The MIP problem `MIP1` minimizes (24) the number of patterns ($t_s^p : \mathbf{h_s^*} = 1$) in the support set of the DNF.

Theorem 2 *The optimal solution* $(\mathbf{u}^*, \mathbf{w}^*, \mathbf{h}^*)$ *of* `MIP1`

$$\min \quad \sum_{s=1}^{M} h_s \tag{24}$$

$$\text{subject to} \quad \sum_{j \in J} \sum_{i=1}^{n_j} \beta_{ij}^k u_{ij,s} + |J| \, (1 - w_s^k) \geq |J|, \qquad k \in \bar{\Omega}_B^+, s = 1, \ldots, M \tag{25}$$

$$\sum_{j \in J} \sum_{i=1}^{n_j} \beta_{ij}^k u_{ij,s} \leq |J| - 1, \qquad k \in \bar{\Omega}_B^-, s = 1, \ldots, M \tag{26}$$

$$w_s^k \leq h_s, \qquad k \in \Omega_B^+, s = 1, \ldots, M \tag{27}$$

$$\sum_{s=1}^{M} w_s^k \geq 1, \qquad k \in \Omega_B^+ \tag{28}$$

$$\sum_{i=1}^{n_j} u_{ij,s} = 1, \qquad j \in J, s = 1, \ldots, M \tag{29}$$

$$0 \leq h_s \leq 1, \qquad s = 1, \ldots, M \tag{30}$$

$$w_s^k \in \{0, 1\}, \qquad k \in \Omega_B^+, s = 1, \ldots, M \tag{31}$$

$$u_{ij,s} \in \{0, 1\}, \qquad i = 1, \ldots, n_j, j \in J, s = 1, \ldots, M \tag{32}$$

defines a DNF

$$f = \bigvee_{\mathbf{h_s^*}=1} t_s^p \tag{33}$$

containing the minimal number of e^p-patterns

$$t_s^p = \bigwedge_{\substack{\mathbf{u_{ij,s}^*}=1, \\ j \in J, i=1,\ldots,n_j}} \beta_{ij}$$

of degree $|J|$ and coverage

$$|Q_s^p| = \left| \left\{ k : \sum_{j \in J} \sum_{i=1}^{n_j} \beta_{ij}^k \, \mathbf{u_{ij,s}^*} = |J|, k \in \bar{\Omega}_B^+ \right\} \right| \tag{34}$$

needed to cover all the e^p- sufficient realizations

$$f(k) \geq 1, \; k \in \Omega_B^+ \, .$$

Proof. Constraints (25) and (26) guarantees that each term t_s^p is a pattern: (25) identifies which $k \in \Omega_B^+$ is covered by t_s^p, while the set of constraints (26) does not allow any $k \in \Omega_B^-$ to be covered by any of the generated terms t_s^p. The minimization of the number of patterns included in f implies that each t_s^p is strong and thus a e^p-pattern (Proposition 2). Constraints (28) ensures that each $k \in \Omega_B^+$ is covered by at least one pattern included in the DNF, since each w_s^k can only take 1 if $t_s^p(k) = 1$ and $h(s) = 1$. Indeed, besides (25) forcing w_s^k to take value 0 if k is not covered by t_s^p, (27) ensures that w_s^k is equal to 0 if t_s^p is not included ($\mathbf{h_s^*} = 0$) in f. Constraint (29) ensures that each term t_s^p contains exactly one literal associated with each component ξ_j. Hence, t_s^p has degree $|J|$. The coverage $|Q_s|$ of t_s^p is equal to the number (34) of $k \in \Omega_B^+$ which satisfy all the conditions imposed by t_s^p. □

Note that we dot need to explicitly impose the decision variables h_s to be binary in `MIP1`. In the optimal solution of `MIP1`, the components of h will naturally take value 0 or 1, when they are constrained to be in interval $[0, 1]$. Since the objective function minimizes $\sum_{s=1}^{M} h_s$, each h_s is set to its minimum (0), if the corresponding e^p-pattern t_s^p is not needed to guarantee the coverage of all e^p-sufficient realizations. On the other hand, if t_s^p is needed to satisfy the coverage condition, at least one of the $w_s^k, k \in \bar{\Omega}_B^-$ takes value 1, which implies (27) that the corrsponding h_s is equal to 1.

The introduction of the set of auxiliary constraints $h_s \geq h_{s+1}, \; s = 1, \ldots, (M-1)$ could facilitate the solution of `MIP1`. Alternatively, we could have the set of constraints $\sum_{k \in \Omega_B^+} w_s^k \geq \sum_{k \in \Omega_B^+} w_{s+1}^{k-1}, s = 1, \ldots, (M-1)$ which would rank the patterns t_s^p in decreasing order of coverage: $|Q_s| \geq |Q_{s+1}|, s = 1, \ldots, (M-1)$ and t_1^p would be the strong e^p-pattern with maximal coverage defined by the optimal solution of `IP1`.

From a computational point of view, it is important to generate a DNF that is as simple as possible [34]. The "simplicity" of a DNF is typically assessed with respect to the degree and the number of the patterns included in the DNF. In that respect, the minimal and prime features are coveted properties. A DNF said to be *minimal* or *irredundant* [14] if the removal of one of its patterns results in a different mapping $\{0, 1\}^n \to \{0, 1\}$, and it is *prime* [14] if all the patterns it includes are prime.

Corollary 1 *The optimal solution of* `MIP1` *defines a prime DNF* $f = \bigvee_{\mathbf{h_s^*}=1} t_s^p$.

It is straightforward, since the DNF only includes e^p-patterns and those are prime by construction.

Corollary 2 *The optimal solution of* `MIP1` *defines a minimal DNF.*

Proof. The optimal solution of `MIP1` defines the smallest cardinality set of patterns for covering every $k \in \Omega_B^+$. It implies that the removal of any one of the patterns t_s^p in f leaves at least one p-sufficient realization uncovered, thus modifying the mapping $\{0,1\}^n \to \{0,1\}$. □

It is well documented [10] that each discrete probability distribution has a finite, yet unknown, number of ps. The solution of `MIP1` answers that question.

Corollary 3 *The number N^p of e^p-patterns and of pLEPs of the probability distribution F at the reliability level p is equal to* $N^p = \sum_{s=1}^{M} \mathbf{h_s^*}$.

4.2 Sequential Construction

The above MIP contains $M \cdot (n + |\Omega_B^+|)$ integer and M continuous decision variables. If the number $|\Omega_B^+|$ of sufficient realizations is high, the solution of the above problem could be challenging. In this section, we develop an alternative, sequential method for the derivation of a DNF containing a set of e^p-patterns covering all p-sufficient realizations. The approach involves the solution of a finite sequence of integer programming problems (i.e., N^p of them) of smaller dimension than `MIP1`. Each iteration s involves three steps: *(i)* solution of an IP problem $\mathtt{IP}_s$ and generation of an e^p-pattern t_s^p; *(ii)* determination of the set Q_s of realizations covered by t_s^p and update of the residual set H_s of uncovered p-efficient realizations; *(iii)* verification of stopping criterion. We denote by H_s the set of p-sufficient realizations that are not covered by any of the e^p-patterns generated at iterations $s' = 1, \ldots, (s-1)$.

A) Initialization: Set $s = 0$.

B) Iterative process:

1. Set $s \leftarrow s + 1$.

   Derivation of an e^p-pattern through the solution of problem $\mathtt{IP}_s$:

$$z = \max \sum_{k \in H_s} w^k \tag{35}$$

$$\text{subject to} \quad \sum_{j \in J} \sum_{i=1}^{n_j} \beta_{ij}^k u_{ij} + |J|\,(1 - w^k) \geq |J|, \qquad k \in H_s \tag{36}$$

$$\sum_{j \in J} \sum_{i=1}^{n_j} \beta_{ij}^k u_{ij} \leq |J| - 1, \qquad k \in \bar{\Omega}_B^- \tag{37}$$

$$\sum_{i=1}^{n_j} u_{ij} = 1, \qquad j \in J \tag{38}$$

$$u_{ij} \in \{0,1\}, \qquad j \in J, i = 1, \ldots, n_j \tag{39}$$

$$w^k \in \{0,1\}, \qquad k \in H_s \tag{40}$$

The optimal solution $(\mathbf{u}^*, \mathbf{w}^*)$ defines the e^p-pattern t_s^p. Go to 2.

At $s = 1$, $H_s = \Omega_B^+$ and $\mathtt{IP}_1 = \mathtt{IP1}$.

2. Determination of coverage $Q_s = \left\{s : \mathbf{w}^{\mathbf{k}^*} = 1, k \in H_s\right\}$ of t_s^p and update of $H_s = H_{s-1} \setminus Q_s$. Go to 3.

3. Verification of stopping criterion. If $H_s = \emptyset$, then stop. Otherwise, go to 1.

It is obvious that the proposed method converges finitely and stops after N^p iterations. The output is a minimal and prime DNF $f = \bigvee_{\mathbf{s=1,\dots,N^p}} t_s^p$ covering all p-sufficient realizations.

5 Illustration

This section illustrates the proposed approach with the numerical example introduced by Prékopa [25] in which the support set of the bivariate random variable ξ is: $\Omega = \{\omega_j : 0 \le \omega_j \le 9, \omega_j \in \mathbb{Z}\},\ j = 1, 2$. The binarization process is carried out with Matlab and the mathematical programming problems are solved with the CPLEX 12.1 solver.

Let $p = 0.6$. The sufficient equivalent set of cut points C^e comprises 12 cut points: n_1=7 and n_2=5: $C^e = \{3, 4, 5, 6, 7, 8, 9\ ;\ 5, 6, 7, 8, 9\}$. The binarization of the set of relevant realizations (containing 29 p- sufficient realizations and 6 p-insufficient ones) and the pdBf associated representing F and $p = 0.6$ are given in Table 1. We derive the prime e^p-pattern with maximal coverage using problem IP1. The problem is solved at the root node and defines the optimal solution $(\mathbf{u}^*, \mathbf{w}^*) = (\mathbf{u}^*, \mathbf{28})$), with $\mathbf{u}^*_{\mathbf{11}} = \mathbf{u}^*_{\mathbf{22}}$ $= 1$, and all the other components of $\mathbf{u}^* = 0$. The prime e^p-pattern with maximal coverage is thus $t^p = \beta_{11}\,\beta_{22}$, and it covers 28 p-sufficient realizations. To generate, using the integrated approach, the minimal and prime DNF composed of e^p-patterns covering all p-sufficient realizations, we solve problem MIP1. The problem is solved in less than 2 seconds. The DNF includes the two (N^p=2) following patterns: $f = \underbrace{\beta_{11}\,\beta_{22}}_{t_1^p} \bigvee \underbrace{\beta_{71}\,\beta_{12}}_{t_2^p}$. Two iterations are needed to generate a prime DNF covering all p-sufficient realizations using the sequential method (Section 4.2).

When $p = 0.8$, C^e comprises 9 cut points: $C^e = \{4, 5, 6, 7, 8, 9\ ;\ 7, 8, 9\}$: n_1=6 and n_2=3. The set of relevant realizations contains 13 p-sufficient realizations and 5 p-insufficient ones. The corresponding pdBf is given in Table 2. The prime e^p-pattern with maximal coverage is $\beta_{11}\,\beta_{22}$. It covers 17 p-sufficient realizations. We obtain the same DNF with both the integrated and the sequential approaches. It contains two patterns: $f = \beta_{11}\,\beta_{22} \bigvee \beta_{61}\,\beta_{12}$.

6 Conclusion

This study revisits the p-efficiency concept introduced by Prékopa [23]. We redefine a pLEP as a so-called e^p-pattern. The new definition uses elements from the combinatorial pattern recognition field [11, 28, 33] and is based on the combinatorial pattern framework for stochastic programming problems proposed in [16]. The method involves the binarization of the probability distribution, and the generation

Table 1: Illustrative Example: p=0.6

	ω^k		β_{ij}												
k	ω_1	ω_2	β_{11}	β_{21}	β_{31}	β_{41}	β_{51}	β_{61}	β_{71}	β_{12}	β_{22}	β_{32}	β_{42}	β_{52}	I^k
1	3	5	0	0	0	0	0	0	0	0	0	0	0	0	0
2	3	6	0	0	0	0	0	0	0	0	0	0	0	0	1
3	3	7	0	0	0	0	0	0	0	0	0	0	0	0	1
4	3	8	0	0	0	0	0	0	0	0	0	0	0	0	1
5	3	9	0	0	0	0	0	0	0	0	0	0	0	0	1
6	4	5	0	0	0	0	0	0	0	0	0	0	0	0	0
7	4	6	0	0	0	0	0	0	0	0	0	0	0	0	1
8	4	7	0	0	0	0	0	0	0	0	0	0	0	0	1
9	4	8	0	0	0	0	0	0	0	0	0	0	0	0	1
10	4	9	0	0	0	0	0	0	0	0	0	0	0	0	1
11	5	5	0	0	0	0	0	0	0	1	0	0	0	0	0
12	5	6	0	0	0	0	0	0	0	1	0	0	0	0	1
13	5	7	0	0	0	0	0	0	0	1	0	0	0	0	1
14	5	8	0	0	0	0	0	0	0	1	0	0	0	0	1
15	5	9	0	0	0	0	0	0	0	1	0	0	0	0	1
16	6	5	0	0	0	0	0	0	0	1	1	0	0	0	0
17	6	6	0	0	0	0	0	0	0	1	1	0	0	0	1
18	6	7	0	0	0	0	0	0	0	1	1	0	0	0	1
19	6	8	0	0	0	0	0	0	0	1	1	0	0	0	1
20	6	9	0	0	0	0	0	0	0	1	1	0	0	0	1
21	7	5	0	0	0	0	0	0	0	1	1	1	0	0	0
22	7	6	0	0	0	0	0	0	0	1	1	1	0	0	1
23	7	7	0	0	0	0	0	0	0	1	1	1	0	0	1
24	7	8	0	0	0	0	0	0	0	1	1	1	0	0	1
25	7	9	0	0	0	0	0	0	0	1	1	1	0	0	1
26	8	5	0	0	0	0	0	0	0	1	1	1	1	0	0
27	8	6	0	0	0	0	0	0	0	1	1	1	1	0	1
28	8	7	0	0	0	0	0	0	0	1	1	1	1	0	1
29	8	8	0	0	0	0	0	0	0	1	1	1	1	0	1
30	8	9	0	0	0	0	0	0	0	1	1	1	1	0	1
31	9	5	0	0	0	0	0	0	0	1	1	1	1	1	1
32	9	6	0	0	0	0	0	0	0	1	1	1	1	1	1
33	9	7	0	0	0	0	0	0	0	1	1	1	1	1	1
34	9	8	0	0	0	0	0	0	0	1	1	1	1	1	1
35	9	9	0	0	0	0	0	0	0	1	1	1	1	1	1

Table 2: Illustrative Example: p=0.8

	ω^k		β_{ij}									
k	ω_1	ω_2	β_{11}	β_{21}	β_{31}	β_{41}	β_{51}	β_{61}	β_{12}	β_{22}	β_{32}	I^k
1	4	7	1	0	0	0	0	0	1	0	0	0
2	4	8	1	0	0	0	0	0	1	1	0	1
3	4	9	1	0	0	0	0	0	1	1	1	1
4	5	7	1	1	0	0	0	0	1	0	0	0
5	5	8	1	1	0	0	0	0	1	1	0	1
6	5	9	1	1	0	0	0	0	1	1	1	1
7	6	7	1	1	1	0	0	0	1	0	0	0
8	6	8	1	1	1	0	0	0	1	1	0	1
9	6	9	1	1	1	0	0	0	1	1	1	1
10	7	7	1	1	1	1	0	0	1	0	0	0
11	7	8	1	1	1	1	0	0	1	1	0	1
12	7	9	1	1	1	1	0	0	1	1	1	1
13	8	7	1	1	1	1	1	0	1	0	0	0
14	8	8	1	1	1	1	1	0	1	1	0	1
15	8	9	1	1	1	1	1	0	1	1	1	1
16	9	7	1	1	1	1	1	1	1	0	0	1
17	9	8	1	1	1	1	1	1	1	1	0	1
18	9	9	1	1	1	1	1	1	1	1	1	1

of a consistent partially defined Boolean function representing the combination (F, p) of the binarized probability distribution F and the enforced probability level p.

To qualify as p-efficient, a point has to satisfy multiple requirements. These can be captured by a combinatorial e^p-pattern which provides a compact representation of the defining characteristics of a pLEP and opens the door to new mathematical approaches for the generation of pLEPs. We show that any pattern representing a pLEP is strong and prime [12] and we propose an integer programming formulation allowing for the generation of an e^p-pattern. We also show that the (finite) collection of pLEPs can be represented as a prime and irreducible DNF and propose an MIP formulation allowing for the construction of the DNF. Its optimal solution defines the cardinality of the set of pLEPs. We also design a sequential method to derive the exhaustive collection of e^p-patterns. We illustrate the proposed approach to a problem studied by Prékopa [25].

References

[1] Alexe G., Hammer P.L. 2006. Spanned Patterns for the Logical Analysis of Data. *Discrete Applied Mathematics* 154 (7), 1039-1049.

[2] Alexe S., Hammer P.L. 2007. Accelerated Algorithm for Pattern Detection in Logical Analysis of Data. *Discrete Applied Mathematics* 154 (7), 1050-1063.

[3] Beraldi P., Ruszczyński A. 2002. A Branch and Bound Method for Stochastic Integer Problems under Probabilistic Constraints. *Optimization Methods and Software* 17 (3), 359-382.

[4] Boros E., Hammer P.L., Ibaraki T., Kogan A. 1997. Logical Analysis of Numerical Data. *Mathematical Programming* 79, 163-190.

[5] Boros E., Hammer P.L., Ibaraki T., Kogan A., Mayoraz E., Muchnik I. 2000. An Implementation of Logical Analysis of Data. *IEEE Transactions on Knowledge and Data Engineering* 12(2), 292-306.

[6] Boros E., Ibaraki T., Makino K. 1997. Monotone Extensions of Boolean Data Sets. In: *Algorithmic Learning Theory*, Lecture Notes in Computer Science. Springer, Berlin-Heidelberg.

[7] Calafiore G.C., Campi M.C. 2005. Uncertain Convex Programs: Randomized Solutions and Confidence Levels. *Mathematical Programming* 102, 25-46.

[8] Charnes A., Cooper W.W., Symonds G.H. 1958. Cost Horizons and Certainty Equivalents: An Approach to Stochastic Programming of Heating Oil. *Management Science* 4, 235-263.

[9] Dentcheva D., Lai B., Ruszczyński A. 2004. Dual Methods for Probabilistic Optimization Problems. *Mathematical Methods of Operations Research* 60 (2), 331-346.

[10] Dentcheva D., Prékopa A., Ruszczyński A. 2001. Concavity and Efficient Points of Discrete Distributions in Probabilistic Programming. *Mathematical Programming* 47 (3), 1997-2009.

[11] Hammer P.L. 1986. Partially Defined Boolean Functions and Cause-Effect Relationships. *International Conference on Multi-Attribute Decision Making Via OR-Based Expert Systems*. University of Passau, Passau, Germany.

[12] Hammer P.L., Bonates T.O. 2006. Logical Analysis of Data: From Combinatorial Optimization to Medical Applications. *Annals of Operations Research* 148 (1), 203-225.

[13] Hammer P.L., Kogan A., Simeone B., Szedmak S. 2004. Pareto-Optimal Patterns in Logical Analysis of Data. *Discrete Applied Mathematics* 144, 79-102.

[14] Ibaraki T. 2010. *Partially Defined Boolean Functions*. In: *Boolean Functions - Theory, Algorithms, and Applications.*. Eds: Crama Y., Hammer P.L. Cambridge Press University. In Press.

[15] Lejeune M.A. 2008. Preprocessing Techniques and Column Generation Algorithms for p-Efficiency. *Journal of Operational Research Society* 59, 1239-1252.

[16] Lejeune M.A. Linear Reformulation of Probabilistically Constrained Optimization Problems Using Combinatorial Patterns. *International Colloquium on Stochastic Modeling and Optimization*. Dedicated to the 80th birthday of Professor András Prékopa. Piscataway, NJ, 2009.

[17] Lejeune M.A., Noyan N. 2010. Mathematical Programming Approaches for Generating e^p- Efficient Points. *Working Paper*.

[18] Lejeune M.A., Ruszczyński A. 2007. An Efficient Trajectory Method for Probabilistic Inventory-Production-Distribution Problems. *Operations Research* 55 (2), 378-394.

[19] Luedtke J., Ahmed S., Nemhauser G. 2010. An Integer Programming Approach for Linear Programs with Probabilistic Constraints. *Mathematical Programming* 122 (2), 247-272.

[20] Martinez-Trinidad J. F., Guzmán-Arenas A. 2001. The Logical Combinatorial Approach to Pattern Recognition, an Overview Through Selected Works. *Pattern Recognition* 34 (4), 741-751.

[21] Prékopa A. 1970. On Probabilistic Constrained Programming. *Proceedings of the Princeton Symposium on Mathematical Programming*. Princeton University Press, 113-138.

[22] Prékopa A. 1973. Contributions to the Theory of Stochastic Programming. *Mathematical Programming* 4, 202-221.

[23] Prékopa A. 1990. Dual Method for a One-Stage Stochastic Programming with Random rhs Obeying a Discrete Probability Distribution. *Zeitschrift of Operations Research* 34, 441-461.

[24] Prékopa A. 1995. *Stochastic Programming*. Kluwer. Boston, MA.

[25] Prékopa A. 2003. Probabilistic Programming Models. Chapter 5 in: *Stochastic Programming: Handbook in Operations Research and Management Science 10*. Eds: Ruszczyński A., Shapiro A. Elsevier Science Ltd, 267-351.

[26] Prékopa A., Vizvari B., Badics T. 1998. Programming under Probabilistic Constraint with Discrete Random Variable. In: *New Trends in Mathematical Programming*. Eds: Giannessi F., Komlósi S., Rapcsák T. Boston, MA.

[27] Radeanu S. 1974. *Boolean Functions and Equations*. North-Holland, Amsterdam.

[28] Ruiz-Shulcloper J., Abidi M.A. 2002. Logical Combinatorial Pattern Recognition: A Review. *Transworld Research Networks* 3, 133-176.

[29] Ruszczyński A. 2002. Probabilistic Programming with Discrete Distribution and Precedence Constrained Knapsack Polyhedra. *Mathematical Programming* 93, 195-215.

[30] Ruszczyński A., Shapiro A. 2003. *Stochastic Programming: Handbook in Operations Research and Management Science* 10. Elsevier Science Ltd.

[31] Ryoo H.S., Jang I.-Y. 2009. MILP Approach to Pattern Generation in Logical Analysis of Data. *Discrete Applied Mathematics*, 157 (4), 749-761.

[32] Torvik V.I., Triantaphyllou E. 2009. Inference of Monotone Boolean Functions. *Encyclopedia of Optimization*, 1591-1598.

[33] Truemper K. 2004. *Design of Logic-Based Intelligent Systems*. Wiley & Sons, Hoboken, NJ.

[34] Urbano R.H., Mueller R.K. 1956. A Topological Method for the Determination of the Minimal Forms of a Boolean Function. *IRE Transactions on Electronic Computers* EC-5, 126-132.