# Solving chance-constrained combinatorial problems to optimality

Olivier Klopfenstein[1,2],
[1]France Télécom R&D, 38-40 rue du gl Leclerc, 92794 Issy-les-Moux cedex 9, France
[2]Université de Technologie de Compiègne, Laboratoire Heudiasyc UMR CNRS 6599,
60205 Compiègne Cedex, France.
olivier.klopfenstein@orange-ftgroup.com

November 27, 2007

**Abstract**

The aim of this paper is to provide new efficient methods for solving general chance-constrained integer linear programs to optimality. Valid linear inequalities are given for these problems. They are proved to characterize properly the set of solutions. They are based on a specific scenario, whose definition impacts strongly on the quality of the linear relaxation built. A branch-and-cut algorithm is described to solve chance-constrained combinatorial problems to optimality. Numerical tests validate the theoretical analysis and illustrate the practical efficiency of the proposed approach.

*keywords:* chance-constrained programming, integer linear programming

## 1 Introduction

Stochastic programming aims at taking account of probabilistic information in mathematical programs. Chance-constrained programming is one of the well-known approaches, where the goal is to find the best solution remaining feasible for a given unfeasibility probability tolerance. While this model was introduced in the fifty's [10], it is still considered as very difficult and widely intractable (see for instance [6, 16, 15]). Among the wide literature on chance-constrained programming, papers on combinatorial problems are not very numerous. The first paper in this field seems to be [14]. The author proposes linear inner and outer approximations to chance constrained problems. 0-1 variables are specifically taken into account, and some branch-and-bound algorithms are described. However, this work is based on specific deterministic equivalent models which are available only with restricted probability assumptions (typically, gaussian-type distributions are needed).

When considering a finite number of scenarios, with associated probabilities, a chance-constrained integer linear problem can be equivalently written as a (standard) integer linear program (see e.g. [13]). This equivalent model, while being natural, is highly intractable. Still with finite discrete scenarios, a branch-and-bound algorithm is proposed in [3] when only the right hand side of constraints is random, with joint probability constraints. When random variables (r.v.s) have special continuous distribution functions, deterministic equivalent models are available. This is well known for normally distributed r.v.s, and we refer to [9] for generalizations of this result. The difficulty is that the obtained equivalent models are non-linear: when dealing with combinatorial problems, we are lead to very hard integer non-linear programs. However, in some cases, these non-linear formulations can be linearized, as observed in [17]. To the best of our knowledge, [23] is the only earlier work aiming at solving general chance-constrained integer linear programs to optimality. The authors proposed a non-standard integer programming approach based on Gröbner basis theory. Unfortunately, without further improvements, the application of this method seems restricted to small-size problems.

A larger number of papers have proposed approximations and heuristic approaches to solve chance-constrained problems. In [4, 2, 20], deterministic robust problems are proposed, whose solutions are known to be feasible for a chance-constrained problem. Only the Bernstein approximation developed in [20] gives some theoretical guarantees for approximating the chance-constrained problem. But these guarantees remain weak, and the complexity of the model makes it hardly usable with integer valued variables. Based on robust optimization models, [18, 17] have proposed iterative heuristics to chance-constrained integer problems. While being of practical interest for obtaining feasible solutions, theoretical guarantees on the solution quality are available only in very simple and restricted cases. Some other methods rely on sampling, and are readily usable with integer variables [8, 11, 19]. Their simplicity makes them very attractive at first sight. However, they may require a very large number of scenarios, and tend to provide too conservative solutions (see [20] for a comparison with the Bernstein approximation). Finally, metaheuristics have also been used to deal with chance-constrained problems, especially with integer variables (see e.g. [1, 5]). In addition to the total lack of theoretical guarantees, using metaheuristics brings some practical difficulties in the tuning of parameters, which may require lots of tests.

In the current paper, the goal is to design branch-and-cut algorithms to solve chance-constrained integer linear programs to optimality. After introducing the general framework of this study in Section 2, valid linear inequalities are described in Section 3. Their definition relies on a basic scenario, whose properties are investigated in Section 4. Part 5 is devoted to algorithmic aspects, and finally numerical tests are reported in Section 6.

# 2 Notations and preliminaries

## 2.1 Main framework

Let $I = \{1, \ldots, n\}$ and $J = \{1, \ldots, m\}$ ($n, m \in \mathbb{N}$). Let us consider a random matrix $A$, whose components $\{A_{ji}\}_{j \in J, i \in I}$ are real-valued random variables (r.v.s), and a vector $b \in \mathbb{R}^m$. In each row $j \in J$, there may be some coefficients which are known with certainty, that is, whose support is restricted to one single value. Hence, we denote by $I_u(j) \subseteq I$ the set of indices corresponding to r.v.s whose support is not restricted to one value. In other words, the $\{A_{ji}\}_{i \in I_u(j)}$ are the uncertain coefficients of row $j$.

Let $\mathbb{P}$ denote the probability measure, we are interested in the following set:

$$X = \Big\{ x \in \{0,1\}^n \mid \forall j \in J, \mathbb{P}(A_j x \leq b_j) \geq 1 - \varepsilon_j \Big\}.$$

For all $j \in J$, $\varepsilon_j \in [0, 1)$ is the unfeasibility tolerance on row $j$. In other words, $X$ is the set of 0-1 points satisfying each constraint $j$ with probability at least $1 - \varepsilon_j$. Note that classical linear constraints are taken into account in this model, since we can have $I_u(j) = \phi$ and $\varepsilon_j = 0$ for some $j \in J$. Considering that only the coefficients of $A$ can be uncertain encompasses clearly the case where the right hand side $b$ is also uncertain. To see that, observe that $A_j x \leq b_j$ can be written with a dummy variable $y$: $A_j x - b_j.y \leq 0$, imposing $y = 1$.

Let $j \in J$, the following notation will be used: $X_j = \Big\{ x \in \{0,1\}^n \mid \mathbb{P}(A_j x \leq b_j) \geq 1 - \varepsilon_j \Big\}$. We have: $X = \bigcap_{j \in J} X_j$.

## 2.2 Extensions of the provided analysis

**Joint probabilistic constraints:** $X$ corresponds to *separate* probabilistic constraints on each row $j \in J$. However, *joint* probabilistic constraints are also important for many real-life applications, where we look for a solution feasible with probability at least $1 - \varepsilon$, $\varepsilon \in [0, 1)$. The associated set of solutions is:

$$X' = \Big\{ x \in \{0,1\}^n \mid \mathbb{P}(Ax \leq b) \geq 1 - \varepsilon \Big\}.$$

**Lemma 1** *If $\varepsilon_j = \varepsilon$ for all $j \in J$, then: $X' \subseteq X$.*

Hence, the characterizations given for $X$ can be used to get relaxations of $X'$.

**Mixed-integer linear programming:** For the sake of simplicity, all the paper is written in a pure integer linear context. However, all the results and algorithms are readily adaptable for mixed integer sets of the following form:

$$X'' = \Big\{ x \geq 0 \mid \forall i \in N, x_i \in \{0,1\}, \text{ and } \forall j \in J, \mathbb{P}(A_j x \leq b_j) \geq 1 - \varepsilon_j \Big\}$$

when: $N \subseteq I$ and $\bigcup_{j \in J} I_u(j) \subseteq N$.

**Non-linear programming:** The approach can be adapted to probabilistic integer non-linear programming. Some more details can be found in Appendix A.

# 3 Describing $X$ with linear inequalities

## 3.1 On classical deterministic equivalent models

In some cases, deterministic equivalent models can be used. Suppose for instance that the uncertain coefficients are independent and normally distributed r.v.s : for each $i \in I_u(j)$, $A_{ji} \sim \mathcal{N}(\mu_{ji}, \sigma_{ji}^2)$. In this case, it is known that (see [22, 24, 6, 15]):

$$X_j = \left\{ x \in \{0,1\}^n \mid \sum_{i \in I_u(j)} \mu_{ji} x_i + \sum_{i \notin I_u(j)} A_{ji} x_i + \Phi^{-1}(1-\varepsilon_j) \sqrt{\sum_{i \in I_u(j)} \sigma_{ji}^2 x_i^2} \leq b_j \right\}$$

where $\Phi$ is the cumulative distribution of the standard normal distribution $\mathcal{N}(0,1)$. Note that similar deterministic equivalent models are available when r.v.s are correlated gaussian r.v.s.

To simplify notations, let us denote: $\mu_{ji} = A_{ji}$ when $i \notin I_u(j)$. The above relation may be transformed in the following way:

$$\begin{array}{ll} & \sum_{i \in I} \mu_{ji} x_i + \Phi^{-1}(1-\varepsilon_j) \sqrt{\sum_{i \in I_u(j)} \sigma_{ji}^2 x_i^2} \leq b_j \qquad (i) \\ \Leftrightarrow & \left\{ \begin{array}{l} \sum_{i \in I} \mu_{ji} x_i \leq b_j, \\ \left[\Phi^{-1}(1-\varepsilon_j)\right]^2 . \sum_{i \in I_u(j)} \sigma_{ji}^2 x_i^2 \leq \left(b_j - \sum_{i \in I} \mu_{ji} x_i\right)^2. \end{array} \right. \end{array}$$

Since $x \in \{0,1\}^n$, the latter expression can be classically linearized. Then (i) is equivalent to:

$$\begin{array}{l} \left\{ \begin{array}{l} \sum_{i \in I} \mu_{ji} x_i \leq b_j, \\ \left[\Phi^{-1}(1-\varepsilon_j)\right]^2 . \sum_{i \in I_u(j)} \sigma_{ji}^2 x_i \leq b_j^2 + \sum_{i \in I} \mu_{ji}(\mu_{ji} - 2b_j) x_i + \sum_{i \neq k} \mu_{ji} \mu_{jk} x_i x_k \end{array} \right. \\ \Leftrightarrow \quad \exists y \geq 0 \text{ s.t. } \left\{ \begin{array}{l} \sum\limits_{i \in I} \mu_{ji} x_i \leq b_j, \\ \left[\Phi^{-1}(1-\varepsilon_j)\right]^2 . \sum\limits_{i \in I_u(j)} \sigma_{ji}^2 x_i \leq b_j^2 + \sum\limits_{i \in I} \mu_{ji}(\mu_{ji} - 2b_j) x_i + \sum\limits_{i \neq k} \mu_{ji} \mu_{jk} y_{ik}, \\ y_{ik} \leq x_i \text{ and } y_{ik} \leq x_k. \end{array} \right. \end{array}$$

Hence, with the above gaussian assumptions, $X_j$ can be represented with a polynomial number of linear inequalities. [9] has proposed interesting extensions from this classical gaussian framework to a more general class of so-called radial distributions. The authors prove that the probabilistic constraint of $X_j$ can be equivalently written as a second-order cone convex constraint. Working with 0-1 variables, such constraints could be linearized as illustrated above.

While the above gaussian framework may seem very restrictive, it provides often an accurate approximation to different probabilistic assumptions. This is due to the well-known central limit theorem, and its variants. However, in the current paper, our aim is to work with hypothesis as general as possible. Such gaussian approximations will not be discussed until the numerical part (Section 6).

## 3.2 Linear inequalities for characterizing $X$

**Definition 1** *Let $j \in J$. We call $\underline{A}_j \in \mathbb{R}^n$ a basic scenario if it satisfies:*

$$\forall I_1 \subseteq I_u(j), I_1 \neq \phi : \mathbb{P}\left(\sum_{i \in I_1} (A_{ji} - \underline{A}_{ji}) \leq 0\right) < 1 - \varepsilon_j. \qquad (1)$$

In the following, such a basic scenario is assumed to be known. From both theoretical and practical viewpoints, this assumption is not restrictive, as it will be explained in Section 4. The central property of a basic scenario is that any $x \in X_j$ has to be feasible for this scenario. This is proved in the next result. For all $i \notin I_u(j)$, the notation $\underline{A}_{ji}$ will sometimes be used instead of $A_{ji}$ to simplify expressions (as in the following statement).

**Lemma 2** *For each $j \in J$, if $\underline{A}_j$ is a basic scenario, then the inequality:*

$$\underline{A}_j x \leq b_j \tag{2}$$

*is valid for $X_j$.*

**Proof:** Suppose that there exists $x \in X_j$ such that $\underline{A}_j x > b_j$, then: $\mathbb{P}(A_j x \leq \underline{A}_j x) \geq \mathbb{P}(A_j x \leq b_j) \geq 1 - \varepsilon_j$. Consider: $I_1 = \{i \in I_u(j) | x_i = 1\}$, we have: $\mathbb{P}\left(\sum_{i \in I_1}(A_{ji} - \underline{A}_{ji}) \leq 0\right) = \mathbb{P}(A_j x \leq \underline{A}_j x) \geq 1 - \varepsilon_j$. As a result, $\underline{A}_j$ is not a basic scenario. □

These inequalities (2) will be said *basic*. Let $j \in J$. For all $i \in I_u(j)$, we denote: $\eta_{ji} = A_{ji} - \underline{A}_{ji}$. Let $\bar{x} \in \{0,1\}^n$, observe that: $\sum_{i \in I_u(j)} A_{ji}\bar{x}_i = \sum_{i \in I_u(j)}(\underline{A}_{ji} + \eta_{ji})\bar{x}_i$. If we introduce: $I_1 = \{i \in I_u(j) | \bar{x}_i = 1\}$, this latter expression may be written: $\sum_{i \in I_u(j)} A_{ji}\bar{x}_i = \sum_{i \in I_u(j)} \underline{A}_{ji}\bar{x}_i + \sum_{i \in I_1} \eta_{ji}$. As a result, we have:

$$\bar{x} \in X_j \Leftrightarrow \mathbb{P}\Big(\sum_{i \in I_1} \eta_{ji} \leq b_j - \underline{A}_j.\bar{x}\Big) \geq 1 - \varepsilon_j.$$

This observation motivates the introduction of the following quantity. For any $I_1 \subseteq I_u(j)$ such that $|I_1| \geq 1$, we define:

$$D_j(I_1) = \sup\Big\{d \mid \mathbb{P}\Big(\sum_{i \in I_1} \eta_{ji} \leq d\Big) < 1 - \varepsilon_j\Big\}. \tag{3}$$

The following properties hold:

**Lemma 3** *Let $j \in J$, let $I_1$ be a non-empty subset of $I_u(j)$:*
*(i) $D_j(I_1) > 0$;*
*(ii) $\mathbb{P}\big(\sum_{i \in I_1} \eta_{ji} \leq D_j(I_1)\big) \geq 1 - \varepsilon_j$.*

**Proof:** Let us start with point (ii). Suppose that: $\mathbb{P}\big(\sum_{i \in I_1} \eta_{ji} \leq D_j(I_1)\big) < 1 - \varepsilon_j$. Since a cumulative distribution is always right-continuous, there exists $\tau > 0$ such that: $\mathbb{P}\Big(\sum_{i \in I_1} \eta_{ji} \leq D_j(I_1) + \tau\Big) < 1 - \varepsilon_j$, that is a contradiction with the definition (3). Then, (i) is a direct consequence of definition (1) and (ii). □

It can also be observed that:

**Lemma 4** *Let $j \in J$. Suppose that the $\{\eta_{ji}\}_{i \in I_u(j)}$ are independent and identically distributed (i.i.d.). Then, for any non-empty subsets $I_1 \subseteq I_u(j)$ and $I_2 \subseteq I_u(j)$ such that $|I_1| = |I_2|$, $D_j(I_1) = D_j(I_2)$.*

This obvious fact will have important practical consequences. Now, these coefficients $D_j(I_1)$ can be used to build valid inequalities for $X_j$:

**Lemma 5** *Let $j \in J$. For any $I_1 \subseteq I_u(j)$ such that $|I_1| \geq 1$, let $I_0 = I_u(j) \setminus I_1$:*

$$\sum_{i \in I_1} \left[\underline{A}_{ji} + D_j(I_1)\right] x_i + \sum_{i \in I_0} \left[\underline{A}_{ji} - D_j(I_1)\right] x_i + \sum_{i \notin I_u(j)} A_{ji} x_i \leq b_j + \Big(|I_1| - 1\Big).D_j(I_1) \tag{4}$$

*is a valid inequality for $X_j$.*

**Proof:** First observe that (4) is equivalently written:

$$\sum_{i \in I} \underline{A}_{ji} x_i \leq b_j + \Big(\sum_{i \in I_0} x_i + |I_1| - \sum_{i \in I_1} x_i - 1\Big).D_j(I_1).$$

Let $d < D_j(I_1)$, let us show that the following inequality is valid for $X_j$:

(a) $\quad \sum_{i \in I} \underline{A}_{ji} x_i \leq b_j + \big(\sum_{i \in I_0} x_i + |I_1| - \sum_{i \in I_1} x_i - 1\big).d.$

Let $x \in X_j$, suppose by contradiction that $x$ violates inequality (a). Consider first the case when $\{i \in I_u(j) | x_i = 1\} = I_1$. Then we have: $\sum_{i \in I} \underline{A}_{ji} x_i > b_j - d$. Then:

$$\mathbb{P}(A_j x \leq b_j) \leq \mathbb{P}\left(A_j x \leq \underline{A}_j x + d\right) = \mathbb{P}\Big(\sum_{i \in I_u(j)} \eta_{ji} x_i \leq d\Big) = \mathbb{P}\Big(\sum_{i \in I_1} \eta_{ji} \leq d\Big).$$

Since $d < D_j(I_1)$, from the definition of $D_j(I_1)$: $\mathbb{P}(\sum_{i \in I_1} \eta_{ji} \leq d) < 1 - \varepsilon_j$, that implies: $\mathbb{P}(A_j x \leq b_j) < 1 - \varepsilon_j$, which is a contradiction.

Consider now the case when $\{i \in I_u(j) | x_i = 1\} \neq I_1$. Then: $\sum_{i \in I_0} x_i + |I_1| - \sum_{i \in I_1} x_i \geq 1$, that implies: $\sum_{i \in I} \underline{A}_{ji} x_i > b_j$ (contradiction, cf Lemma 2).

Hence, inequality (a) is proved to be valid for $X_j$ for all $d < D_j(I_1)$, and thus (a) is valid also for $D_j(I_1)$ itself. □

In the following result, inequalities (4) are shown to characterize exactly the set of 0-1 points in $X$:

**Proposition 1**

$$X = \Big\{ x \in \{0,1\}^n \mid x \textit{ satisfies all inequalities } (4) \Big\}$$

**Proof:** The inclusion $\subseteq$ comes from Lemma 5. For the reverse inclusion, consider $x \in \{0,1\}^n$ such that $x \notin X$: there is $j \in J$ such that $\mathbb{P}(A_j x \leq b_j) < 1 - \varepsilon_j$. Suppose that $x$ satisfies all inequalities (4): in particular, we have $\underline{A}_j x + D_j(I_1) \leq b_j$, with $I_1 = \{i \in I_u(j) | x_i = 1\}$. Hence, we deduce that: $\mathbb{P}(A_j x \leq \underline{A}_j x + D_j(I_1)) < 1 - \varepsilon_j$. This is equivalent to: $\mathbb{P}(\eta_j x \leq D_j(I_1)) = \mathbb{P}\big(\sum_{i \in I_1} \eta_{ji} \leq D_j(I_1)\big) < 1 - \varepsilon_j$, that is contradictory with point (ii) of Lemma 3. □

Some remarks have to be done. First, from the above proposition, the set $X$ can then be described with $\sum_{j \in J} (2^{|I_u(j)|} - 1)$ linear inequalities. However, it is clear that we never need more than $2^{|I_u|}$ inequalities to characterize $X$, where $I_u = \bigcup_{j \in J} I_u(j)$. Indeed, for each $x \in \{0,1\}^n$ such that $x \notin X$, one single inequality from (4) is necessary to separate $x$ from $X$. Its choice only involves the components $\{x_i\}_{i \in I_u}$.

Secondly, from the proof of Lemma 5, we see that inequality (4) remains valid for $X$ when considering $d \leq D_j(I_1)$ instead of $D_j(I_1)$. However, the inequality written with $D_j(I_1)$ is stronger than another written with $d < D_j(I_1)$, when $\{i \in I_u(j) | x_i = 1\} = I_1$. When $|I_1| - \sum_{i \in I_1} x_i - 1 \geq 0$, the inequalities written with $D_j(I_1)$ and $d < D_j(I_1)$ are both dominated by the basic inequality $\underline{A}_j x \leq b_j$.

Finally, note that even though inequalities (4) are sufficient to define the set of 0-1 points in $X$, they do not dominate inequalities (2) in general. As a result, the basic inequalities (2) should also be used in practice to strengthen the continuous relaxation of $X$.

# 4 On basic scenarios

## 4.1 Structure of the set of basic scenarios

Let $j \in J$, let $\mathcal{B}_j$ denote the set of basic scenarios, i.e. the set of vectors satisfying property (1). It is clear that any element in its closure $cl(\mathcal{B}_j)$ can in practice be used as a basic scenario to build inequalities (4).

**Lemma 6** *Let $j \in J$, $cl(\mathcal{B}_j)$ is a polyhedron.*

**Proof:** For all $I_1 \subseteq I_u(j)$ non-empty, we introduce the r.v.: $S(I_1) = \sum_{i \in I_1} A_{ji}$. Then, $\underline{A}_j$ is a basic scenario if, and only if:

$$\forall I_1 \subseteq I_u(j), I_1 \neq \phi : \sum_{i \in I_1} \underline{A}_{ji} < \sup\big\{s \mid \mathbb{P}(S(I_1) \leq s) < 1 - \varepsilon_j\big\}. \tag{5}$$

Then, the set $cl(\mathcal{B}_j)$ is characterized by a finite number of linear inequalities. □

The scenarios in $cl(\mathcal{B}_j)$, which cannot be expressed as the linear combination of two other scenarios in $cl(\mathcal{B}_j)$, are said *extreme*.

**Lemma 7** *The linear description of $X_j$ obtained from all basic scenarios of $cl(\mathcal{B}_j)$ is equivalent to this obtained when considering only extreme ones.*

**Proof:** Consider a non-extreme basic scenario $\underline{A}_j \in \mathcal{B}_j$, there exist two other distinct basic scenarios $\underline{A}_j^1$ and $\underline{A}_j^2$ such that: $\underline{A}_j = \lambda \underline{A}_j^1 + (1-\lambda)\underline{A}_j^2$, with $\lambda \in (0,1)$. Let $I_1 \subseteq I_u(j)$ be non-empty. Observe that:

$$D_j(I_1) + \sum_{i \in I_1} \underline{A}_{ji} = D_j^1(I_1) + \sum_{i \in I_1} \underline{A}_{ji}^1 = D_j^2(I_1) + \sum_{i \in I_1} \underline{A}_{ji}^2 \tag{6}$$

since these quantities are all equal to: $\sup\{s \mid \mathbb{P}\big(\sum_{i\in I_1} A_{ji} \le s\big) < 1 - \varepsilon_j\}$. We deduce that: $D_j(I_1) = \lambda D_j^1(I_1) + (1-\lambda) D_j^2(I_1)$. Then, it is clear that the inequality (4) obtained from $\underline{A}_j$ is a linear combination of those corresponding to $\underline{A}_j^1$ and $\underline{A}_j^2$. □

Unfortunately, extreme basic scenarios are most of the time very numerous and hard to compute.

## 4.2 Building good basic scenarios

The following result can be used to obtain a first basic scenario $\underline{A}_j$, depending on the specific probabilistic assumptions made:

**Lemma 8** *Let $j \in J$, let us denote: $n_j = |I_u(j)|$. Suppose that $\underline{A}_j$ is defined as follows:*
*(i) when no particular probabilistic assumption is made, for all $i \in I_u(j)$: $\mathbb{P}(A_{ji} \le \underline{A}_{ji}) < (1-\varepsilon_j)/n_j$;*
*(ii) if the r.v.s $\{A_{ji}\}_{i \in I_u(j)}$ are independent, for all $i \in I_u(j)$: $\mathbb{P}(A_{ji} \le \underline{A}_{ji}) < 1 - \varepsilon_j^{1/n_j}$;*
*(iii) if the r.v.s $\{A_{ji}\}_{i\in I_u(j)}$ are independent and symmetrically distributed, and if $\varepsilon_j < 0.5$, for all $i \in I_u(j)$: $\underline{A}_{ji} = \mathbb{E}[A_{ji}]$.*

*Then $\underline{A}_j$ is a basic scenario, i.e. it satisfies* (1)*.*

**Proof:** Let $I_1 \subseteq I_u(j)$. We have:

$$\mathbb{P}\Big(\textstyle\sum_{i\in I_1}(A_{ji} - \underline{A}_{ji}) \le 0\Big) \le \mathbb{P}\Big(\exists i \in I_1 \text{ s.t. } A_{ji} \le \underline{A}_{ji}\Big) \le \mathbb{P}\Big(\exists i \in I_u(j) \text{ s.t. } A_{ji} \le \underline{A}_{ji}\Big).$$

When no particular assumption is made, we always have: $\mathbb{P}(\exists i \in I_u(j) \text{ s.t. } A_{ji} \le \underline{A}_{ji}) \le \sum_{i \in I_u(j)} \mathbb{P}(A_{ji} \le \underline{A}_{ji})$. Assuming that (i) holds, this ensures: $\mathbb{P}\big(\sum_{i\in I_1}(A_{ji} - \underline{A}_{ji}) \le 0\big) < n_j.(1-\varepsilon_j)/n_j = 1 - \varepsilon_j$.
Suppose now that r.v.s are independent. Observe that: $\mathbb{P}(\exists i \in I_u(j) \text{ s.t. } A_{ji} \le \underline{A}_{ji}) = 1 - \mathbb{P}(\forall i \in I_u(j), A_{ji} > \underline{A}_{ji}) = 1 - \prod_{i\in I_u(j)} \mathbb{P}(A_{ji} > \underline{A}_{ji})$. From (ii), we deduce: $\mathbb{P}\big(\sum_{i \in I_1}(A_{ji} - \underline{A}_{ji}) \le 0\big) < 1 - \varepsilon_j$.
Finally, suppose that the r.v.s are independent and symmetrically distributed. In this case, the r.v. $S = \sum_{i\in I_1} A_{ji}$ is also symmetrically distributed. This implies that: $\mathbb{P}(S \le \mathbb{E}[S]) = 0.5$. Then, from (iii): $\mathbb{P}\big(\sum_{i\in I_1}(A_{ji} - \underline{A}_{ji}) \le 0\big) = \mathbb{P}(\sum_{i\in I_1} A_{ji} \le \sum_{i \in I_1} \underline{A}_{ji}) = \mathbb{P}(S \le \mathbb{E}[S]) = 0.5$. It proves that (1) holds when $\varepsilon_j < 0.5$. □

The linear description of $X_j$ given in Proposition 1 is based on a basic scenario $\underline{A}_j$, which is not unique. We show that the choice of this basic scenario has a direct impact on the strength of inequalities (2) and (4):

**Lemma 9** *Let $j \in J$, consider two basic scenarios $\underline{A}_j^1$ and $\underline{A}_j^2$ such that $\underline{A}_j^1 \le \underline{A}_j^2$. Then, the linear relaxation of $X_j$ obtained from inequalities* (2) *and* (4) *with $\underline{A}_j^2$ is stronger than this associated with $\underline{A}_j^1$.*

**Proof:** Consider first basic inequalities (2): since $\underline{A}_j^1 \le \underline{A}_j^2$, it is clear that $\underline{A}_j^2 x \le b_j$ dominates $\underline{A}_j^1 x \le b_j$. Now, let us focus on inequalities (4). Let any $I_1 \subseteq I_u(j)$, $I_1 \neq \phi$. Let $D_j^1(I_1)$, $D_j^2(I_1)$, associated with $\underline{A}_j^1$, $\underline{A}_j^2$. Remembering the observation (6), we have: $D_j^1(I_1) = D_j^2(I_1) + \sum_{i \in I_1}(\underline{A}_{ji}^2 - \underline{A}_{ji}^1)$. Let $x \in [0,1]^n$, and let us introduce the following notations (as before, $I_0 = I_u(j) \setminus I_1$):

$$\begin{cases} Q(x) = \sum_{i\in I_1} x_i - \sum_{i \in I_0} x_i - |I_1| + 1, \\ \Gamma_j^1(x) = \sum_{i\in I} \underline{A}_{ji}^1 x_i + Q(x).D_j^1(I_1), \\ \Gamma_j^2(x) = \sum_{i\in I} \underline{A}_{ji}^2 x_i + Q(x).D_j^2(I_1). \end{cases}$$

Then:

$$\begin{array}{rcl}\Gamma_j^1(x) & = & \sum_{i\in I} \underline{A}_{ji}^1 x_i + Q(x).\left[D_j^2(I_1) + \sum_{i\in I_1}(\underline{A}_{ji}^2 - \underline{A}_{ji}^1)\right] \\ & = & \Gamma_j^2(x) + \sum_{i\in I}(\underline{A}_{ji}^1 - \underline{A}_{ji}^2)x_i + Q(x).\sum_{i\in I_1}(\underline{A}_{ji}^2 - \underline{A}_{ji}^1) \\ & = & \Gamma_j^2(x) + \sum_{i\in I_1}(\underline{A}_{ji}^1 - \underline{A}_{ji}^2)\,(x_i - Q(x)) + \sum_{i\in I_0}(\underline{A}_{ji}^1 - \underline{A}_{ji}^2)x_i.\end{array}$$

Suppose that there exists $i \in I_1$ such that $x_i < Q(x)$. In this case, we would have: $\sum_{k\in I_1\setminus\{i\}} x_k - \sum_{i\in I_0} x_i + 1 - |I_1| > 0$, i.e.: $\sum_{k\in I_1\setminus\{i\}} x_k - \sum_{i\in I_0} x_i > |I_1| - 1$. Since $x_k \leq 1$ for all $k \in I_1$, this is not possible. Hence, it is proved that: $\forall i \in I_1, x_i \geq Q(x)$.

Finally, remembering that $\underline{A}_j^1 \leq \underline{A}_j^2$ and $x \geq 0$, we obtain that: $\Gamma_j^1(x) \leq \Gamma_j^2(x)$. Then, we deduce that for all $x \in [0,1]^n$: $\Gamma_j^2(x) \leq b_j \Rightarrow \Gamma_j^1(x) \leq b_j$. That shows that the linear relaxation of $X_j$ obtained from $\underline{A}_j^2$ is stronger than this associated with $\underline{A}_j^1$. □

Hence, we have interest in taking a basic scenario as large as possible. A basic scenario $\underline{A}_j \in cl(\mathcal{B}_j)$ is said *non-dominated* when it cannot be dominated by another one, i.e.: $\forall A_j \in cl(\mathcal{B}_j), A_j \geq \underline{A}_j \Rightarrow A_j = \underline{A}_j$. Given a dominated basic scenario $\underline{A}_j$, we may try to improve it:

**Lemma 10** *Let $j \in J$, let $\underline{A}_j$ be the current basic scenario. Let $v \in \mathbb{R}^n$ such that $v_i > 0$ for all $i \in I_u(j)$, and $v_i = 0$ otherwise. Then $\underline{A}_j + \lambda^*.v \in cl(\mathcal{B}_j)$, with:*

$$\lambda^* = \min_{I_1\subseteq I_u(j), I_1\neq\phi} \left\{ \frac{D_j(I_1)}{\sum_{i\in I_1} v_i} \right\}.$$

*Furthermore, if: $I_u(j) \in \arg\min_{I_1\subseteq I_u(j)} \left\{D_j(I_1)/\sum_{i\in I_1} v_i\right\}$, then $\underline{A}_j + \lambda^*.v$ is non-dominated.*

**Proof:** From the proof of Lemma 6, each subset $I_1$ is associated with an hyperplane whose equation is: $\sum_{i\in I_1} a_{ji} = \sup\left\{s \mid \mathbb{P}\left(\sum_{i\in I_1} A_{ji} \leq s\right) < 1 - \varepsilon_j\right\} = D_j(I_1) + \sum_{i\in I_1} \underline{A}_{ji}$ (cf (5)). Let us denote by $\mathcal{L}$ the straight line of scenarios equal to $\underline{A}_j + \lambda.v$ for some $\lambda \in \mathbb{R}$. Let $I_1 \subseteq I_u(j)$ be non-empty, and let us denote: $\lambda(I_1) = D_j(I_1)/\sum_{i\in I_1} v_i \geq 0$. Then, observe that $\underline{A}'_j = \underline{A}_j + \lambda(I_1).v$ belongs to the intersection of $\mathcal{L}$ with the hyperplane associated with $I_1$, since: $\sum_{i\in I_1} \underline{A}'_{ji} = D_j(I_1) + \sum_{i\in I_1} \underline{A}_{ji}$. Then, it comes that $\lambda^* = \min_{I_1\subseteq I_u(j)} \lambda(I_1)$ corresponds to a scenario in $cl(\mathcal{B}_j)$.

Suppose that $I_u(j) \in \arg\min\left\{D_j(I_1)/\sum_{i\in I_1} v_i\right\}$. This implies that: $\sum_{i\in I_u(j)} \underline{A}'_{ji} = D_j(I_u(j)) + \sum_{i\in I_u(j)} \underline{A}_{ji}$. Then, none of the coefficients $\{\underline{A}'_{ji}\}_{i\in I_u(j)}$ can be increased without violating an inequality valid for $\mathcal{B}_j$. □

Computing $\lambda^*$ requires exponential time in general. To remedy this practical difficulty, we look for another coefficient $\lambda \leq \lambda^*$ easier to obtain. In most of the practical problems, given a direction $v$ of improvement, the following quantity is easy to compute for each $k \in \{1, \ldots, |I_u(j)|\}$:

$$\lambda(k) = \frac{\min\left\{D_j(I_1) \mid I_1 \subseteq I_u(j), |I_1| = k\right\}}{\max\left\{\sum_{i\in I_1} v_i \mid I_1 \subseteq I_u(j), |I_1| = k\right\}}\,. \tag{7}$$

Then, it is clear that: $\min_k \lambda(k) \leq \lambda^*$. This value can be used to improve the initial basic scenario in a given direction $v$.

**Lemma 11** *Let $j \in J$, suppose that the $\{\eta_{ji}\}_{i\in I_u(j)}$ are i.i.d.: $\lambda^* = \min_k \lambda(k)$.*

Indeed, in this case, $D_j(I_1)$ only depends on the cardinality $|I_1|$ (cf Lemma 4): $\lambda(k) = \min_{I_1:|I_1|=k}\left\{D_j(I_1)/\sum_{i\in I_1} v_i\right\}$. In this setting of i.i.d. r.v.s, the numerator of $\lambda(k)$ is obtained directly by computing one value $D_j$, for each $k$; hence, due to the denominators of $\{\lambda(k)\}_k$, $\lambda^*$ can be computed in time $O(n\log(n))$ (sorting complexity of components $\{v_i\}_{i\in I_u(j)}$).

**Lemma 12** *Let $j \in J$, suppose that the $\{\eta_{ji}\}_{i\in I_u(j)}$ are i.i.d.. Consider the direction $v$ such that $v_i = 1$ for all $i \in I_u(j)$, and $v_i = 0$ elsewhere. Then, $\underline{A}_j^* = \underline{A}_j + \lambda^*.v$ is a non-dominated scenario in $cl(\mathcal{B}_j)$.*

**Proof:** That $\underline{A}_j^* \in cl(\mathcal{B}_j)$ comes from the previous lemma. Suppose that this scenario is dominated. There exist $k \in I_u(j)$ and $\underline{A}'_j \in cl(\mathcal{B}_j)$ such that: $\underline{A}'_{jk} > \underline{A}^*_{jk}$ and $\underline{A}'_{ji} = \underline{A}^*_{ji}$ for all $i \neq k$. The idea of the proof is quite simple: since r.v.s are i.i.d. and $v_i = 1$ for all $i \in I_u(j)$, all data are symmetrical, all directions play the same role. As a result, this domination in the direction $k$ implies the existence of a domination in all directions $l \neq k$, $l \in I_u(j)$. Taking a convex combination of the $|I_u(j)|$ resulting scenarios leads to a domination in direction $v$, which is a contradiction with the definition of $\lambda^*$.

Let us detail these ideas. Consider any $l \in I_u(j) \setminus \{k\}$. Let us introduce another scenario $\underline{A}_j^{(l)}$ equal to $\underline{A}_j^*$, except: $\underline{A}_{jl}^{(l)} = \underline{A}_{jl}^* + \underline{A}'_{jk} - \underline{A}_{jk}^*$. Hence: $\underline{A}_{jl}^{(l)} > \underline{A}_{jl}^*$.

Now, our goal is to show that $\underline{A}_j^{(l)} \in cl(\mathcal{B}_j)$. Let $I_1 \subseteq I_u(j)$, we are interested in the r.v.: $S^{(l)}(I_1) = \sum_{i \in I_1} A_{ji} - \underline{A}_{ji}^{(l)}$. Similarly, we denote: $S'(I_1) = \sum_{i \in I_1} A_{ji} - \underline{A}'_{ji}$. Let us prove that $S^{(l)}(I_1) = S'(I_1)$ for all non-empty $I_1 \subseteq I_u(j)$. If $l \notin I_1$ and $k \notin I_1$, this is clear. Consider now the case when $l \in I_1$ and $k \in I_1$. We have: $S^{(l)}(I_1) = \sum_{i \in I_1 \setminus \{k,l\}} (A_{ji} - \underline{A}'_{ji}) + A_{jl} - \underline{A}_{jl}^{(l)} + A_{jk} - \underline{A}_{jk}^{(l)}$. Furthermore: $A_{jl} - \underline{A}_{jl}^{(l)} + A_{jk} - \underline{A}_{jk}^{(l)} = A_{jl} - \underline{A}_{jl}^* - \underline{A}'_{jk} + \underline{A}_{jk}^* + A_{jk} - \underline{A}_{jk}^* = A_{jl} - \underline{A}'_{jl} + A_{jk} - \underline{A}'_{jk}$. As a result, we have: $S^{(l)}(I_1) = S'(I_1)$.

Suppose now that $l \in I_1$ and $k \notin I_1$. We have: $S^{(l)}(I_1) = \sum_{i \in I_1 \setminus \{l\}} (A_{ji} - \underline{A}'_{ji}) + A_{jl} - \underline{A}_{jl}^{(l)}$. Observe that, since $\eta_{jl}$ and $\eta_{jk}$ are i.i.d.: $A_{jl} - \underline{A}_{jl}^{(l)} = \eta_{jl} + \underline{A}_{jl} - \underline{A}_{jl}^{(l)} = \eta_{jk} + \underline{A}_{jl} - \underline{A}_{jl}^* - \underline{A}'_{jk} + \underline{A}_{jk}^* = \eta_{jk} - \underline{A}'_{jk} - \lambda^* + \underline{A}_{jk}^* = \eta_{jk} - \underline{A}'_{jk} + \underline{A}_{jk} = A_{jk} - \underline{A}'_{jk}$. Then, it is proved that: $S^{(l)}(I_1) = S'\big((I_1 \setminus \{l\}) \cup \{k\}\big)$. Furthermore, since the r.v.s $\{\eta_{ji}\}_{i \in I_u(j)}$ have been assumed i.i.d., and since $|I_1| = |(I_1 \setminus \{l\}) \cup \{k\}|$, we have: $S'\big((I_1 \setminus \{l\}) \cup \{k\}\big) = S'(I_1)$.

Finally, consider the case when $l \notin I_1$ and $k \in I_1$. We have: $S^{(l)}(I_1) = \sum_{i \in I_1 \setminus \{k\}} (A_{ji} - \underline{A}'_{ji}) + A_{jk} - \underline{A}_{jk}^{(l)}$. Since $\eta_{jl}$ and $\eta_{jk}$ are i.i.d., we also have: $A_{jk} - \underline{A}_{jk}^{(l)} = \eta_{jk} + \underline{A}_{jk} - \underline{A}_{jk}^{(l)} = \eta_{jl} + \underline{A}_{jk} - \underline{A}_{jk}^* = A_{jl} - \underline{A}_{jl} - \lambda^* = A_{jl} - \underline{A}_{jl}^* = A_{jl} - A'_{jl}$. Hence: $S^{(l)}(I_1) = S'\big((I_1 \setminus \{k\}) \cup \{l\}\big)$. As before, since $|I_1| = |(I_1 \setminus \{k\}) \cup \{l\}|$, we have: $S'\big((I_1 \setminus \{k\}) \cup \{l\}\big) = S'(I_1)$.

Hence, it is proved that, for all non-empty $I_1 \subseteq I_u(j)$: $S^{(l)}(I_1) = S'(I_1)$. As a result, since $\underline{A}'_j \in cl(\mathcal{B}_j)$, it is shown that $\underline{A}_j^{(l)} \in cl(\mathcal{B}_j)$ also (cf property (1)). This is true for all $l \in I_u(j) \setminus \{k\}$. Then, consider the convex combination of the built scenarios: $\underline{A}''_j = 1/n_j.\big(\underline{A}'_j + \sum_{l \in I_u(j) \setminus \{k\}} \underline{A}_j^{(l)}\big)$, with $n_j = |I_u(j)|$. We have:

$$\underline{A}''_j = \underline{A}_j^* + \frac{\underline{A}'_{jk} - \underline{A}_{jk}^*}{n_j}.v \ .$$

This means that: $\underline{A}''_j = \underline{A}_j + \lambda''.v$, with $\lambda'' > \lambda^*$. That is a contradiction with the definition of $\lambda^*$. □

In our practical tests, even when the $\{\eta_{ji}\}_{i \in I(j)}$ are not i.i.d., the direction $v$ proposed in the above lemma appeared to be very efficient when working with the approximation (7). We observed that taking vectors with "less balanced" components penalizes a lot $\lambda(1)$, which was often kept as the minimal value ($\min_k \lambda(k) = \lambda(1)$).

## 4.3 Illustration on a simple example

In this section, let us focus on the set $X = \Big\{x \in \{0,1\}^n \mid \mathbb{P}(a_1 x_1 + a_2 x_2 \leq 2) \geq 4/5\Big\}$, where $a_1$ and $a_2$ are two independent random variables uniformly distributed respectively on [0,1] and on [1,2].

The most obvious basic scenario for this problem is $\underline{a} = (0,1)$, composed with the lower values of each r.v.. The corresponding basic inequality is: $0.x_1 + 1.x_2 = x_2 \leq 2$. Note that with the basic scenario $\underline{a}$, $\eta_1 = a_1 - \underline{a}_1$ and $\eta_2 = a_2 - \underline{a}_2$ are i.i.d.. We enumerate all inequalities (4) for $X$:

- when $I_1 = \{1\}$: we have: $\mathbb{P}(\eta_1 \leq d) < 4/5 \Leftrightarrow d < 4/5$. As a result: $D(I_1) = 4/5$. The corresponding valid inequality is then: $(0 + 4/5).x_1 + (1 - 4/5).x_2 \leq 2$, i.e.: $4x_1 + x_2 \leq 10$.

- When $I_1 = \{2\}$: we have: $\mathbb{P}(\eta_2 \leq d) < 4/5 \Leftrightarrow d < 4/5$. As before, we derive the corresponding valid inequality: $(0 - 4/5).x_1 + (1 + 4/5).x_2 \leq 2$, i.e.: $-4x_1 + 9x_2 \leq 10$.

- When $I_1 = \{1,2\}$: we have, by classical computations: $\mathbb{P}(\eta_1 + \eta_2 \leq d) < 4/5 \Leftrightarrow d < 2 - \sqrt{2/5}$. Then: $D(I_1) = 2 - \sqrt{2/5} \geq 1.36$. The corresponding inequality is: $(0 + D(I_1)).x_1 + (1 + D(I_1)).x_2 \leq 2 + (2-1).D(I_1)$, i.e.: $1.36.x_1 + 2.36.x_2 \leq 3.36$.

Observe that the first two inequalities are implied by the domain constraints $x_{1,2} \leq 1$: they are useless.

**Impact of the basic scenario:** As r.v.s are independent and symmetrically distributed, from Lemma 8, the basic scenario can in fact be defined as: $\underline{a}' = (1/2, 3/2)$. The corresponding basic inequality is: $1/2.x_1 + 3/2.x_2 \leq 2$ (or equivalently: $x_1 + 3x_2 \leq 4$). As before, we give the inequalities (4):

- when $I_1 = \{1\}$: we have: $\mathbb{P}(\eta_1 \leq d) < 4/5 \Leftrightarrow d < 4/5 - 1/2 = 3/10$. As a result: $D(I_1) = 3/10$, and the corresponding valid inequality is: $4/5.x_1 + 6/5.x_2 \leq 2$, i.e.: $4x_1 + 6x_2 \leq 10$.
- When $I_1 = \{2\}$: similarly: $\mathbb{P}(\eta_2 \leq d) < 4/5 \Leftrightarrow d < 3/10$. As before, we derive the corresponding valid inequality: $1/5.x_1 + 9/5.x_2 \leq 2$, i.e.: $x_1 + 9x_2 \leq 10$.
- When $I_1 = \{1, 2\}$: we have, by classical computations: $\mathbb{P}(\eta_1 + \eta_2 \leq d) < 4/5 \Leftrightarrow d < 1 - \sqrt{2/5}$. Then: $D(I_1) = 1 - \sqrt{2/5} \geq 0.36$, and we obtain: $0.86.x_1 + 1.86.x_2 \leq 2.36$.

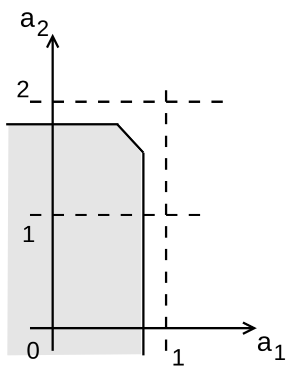
As intended, the obtained inequalities are stronger than those built from the basic scenario $\underline{a}$, since $\underline{a}' \geq \underline{a}$.

**Description of basic scenarios:** We give a complete description of basic scenarios. Let us remark that:

$$\begin{cases} \mathbb{P}(a_1 \leq \alpha) < 4/5 \Leftrightarrow \alpha < 4/5, \\ \mathbb{P}(a_2 \leq \alpha) < 4/5 \Leftrightarrow \alpha < 9/5, \\ \mathbb{P}(a_1 + a_2 \leq \alpha) < 4/5 \Leftrightarrow \alpha < 3 - \sqrt{2/5}. \end{cases}$$

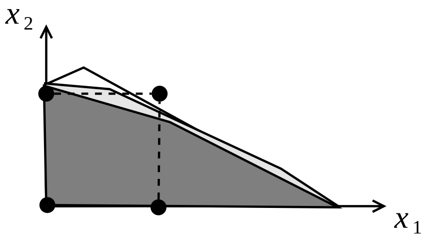
Then, the set of basic scenarios is:

$$cl(\mathcal{B}) = \left\{ \tilde{a} \in \mathbb{R}^2 \mid \tilde{a}_1 \leq 4/5, \tilde{a}_2 \leq 9/5 \text{ and } \tilde{a}_1 + \tilde{a}_2 \leq 3 - \sqrt{2/5} \right\}.$$




Finally, we use Lemma 10 to improve the naive basic scenario $\underline{a}$ in the direction $v = (1, 1)$ (cf Lemma 12). From easy calculations, we obtain: $\lambda^* = 1 - 1/\sqrt{10}$, and then the new basic scenario is: $\underline{a}'' = (1 - 1/\sqrt{10}, 2 - 1/\sqrt{10})$.

The three linear relaxations respectively obtained from basic scenarios $\underline{a}$, $\underline{a}'$ and $\underline{a}''$ are represented on the following figure:



# 5 A solution algorithm for chance-constrained combinatorial problems

Let us focus now on the following optimization problem, called Chance-Constrained Integer Linear Problem (CCILP):

$$\max \Big\{ cx \mid x \in X \Big\}$$

where $c \in \mathbb{R}^n$. The previous section provides a linear relaxation of $X$ which, in theory, makes a classical branch-and-bound algorithm possible. The difficulty lies in the exponential number of inequalities (4). They cannot be all present together in the implemented model and have to be generated in a suitable way.

## 5.1 Separation of inequalities (4)

In this section, let $\tilde{x} \in [0,1]^n$ (possibly fractional). Let $j \in J$, for any subset $I_1 \subseteq I_u(j)$, $I_0 = I_u(j) \setminus I_1$, we denote:

$$\begin{aligned} \Delta_j(I_1) = \quad & b_j + \Big(|I_1| - 1\Big).D_j(I_1) \\ & - \left\{ \sum_{i \in I_1} \left[\underline{A}_{ji} + D_j(I_1)\right] \tilde{x}_i + \sum_{i \in I_0} \left[\underline{A}_{ji} - D_j(I_1)\right] \tilde{x}_i + \sum_{i \notin I_u(j)} A_{ji} \tilde{x}_i \right\}. \end{aligned} \tag{8}$$

$\Delta_j(I_1)$ is the slack value of the inequality (4) associated with $j$ and $I_1$ for $\tilde{x}$. The goal is to find the subset $I_1$ corresponding to the minimum slack possible (most violated inequality for $\tilde{x}$). In general, this cannot be done directly from the above formula, since the complete enumeration of subsets is not tractable. First, remember that each inequality associated with $I_1$ is specially related to the point $x(I_1)$, which is the characteristic vector of $I_1$ (see Section 3.2). Then, we look for $I_1$ such that $x(I_1)$ is "close to" $\tilde{x}$. As a result, all indices $i \in I_u(j)$ such that $\tilde{x}_i = 1$ should appear in $I_1$.

Let us describe now a way of handling the fractional components of $\tilde{x}$. A natural idea is to extend $I_1$ in an iterative way. Consider a fixed $I_1 \subseteq I_u(j)$, and let us introduce $I_1' = I_1 \cup \{k\}$ for some $k \in I_u(j) \setminus I_1$. We have:

$$\begin{aligned} \Delta_j(I_1') - \Delta_j(I_1) &= (|I_1| - 1).\big[D_j(I_1') - D_j(I_1)\big] + D_j(I_1') \\ &\quad - \big[D_j(I_1') - D_j(I_1)\big].\big(\textstyle\sum_{i \in I_1} \tilde{x}_i - \sum_{i \in I_0 \setminus \{k\}} \tilde{x}_i\big) - \big[D_j(I_1') + D_j(I_1)\big].\tilde{x}_k \\ &= \big[D_j(I_1') - D_j(I_1)\big].\Big[|I_1| - 1 - \textstyle\sum_{i \in I_1} \tilde{x}_i + \sum_{i \in I_0 \setminus \{k\}} \tilde{x}_i\Big] \\ &\quad + D_j(I_1') - \big[D_j(I_1') + D_j(I_1)\big].\tilde{x}_k \ . \end{aligned}$$

**Lemma 13** *We have:*

*(i) if $D_j(I_1') \geq D_j(I_1)$: $\Delta_j(I_1') - \Delta_j(I_1) \geq D_j(I_1) - \big[D_j(I_1') + D_j(I_1)\big].\tilde{x}_k$,*

*(ii) if $D_j(I_1') \leq D_j(I_1)$: $\Delta_j(I_1') - \Delta_j(I_1) \leq D_j(I_1) - \big[D_j(I_1') + D_j(I_1)\big].\tilde{x}_k$,*

*(iii) if: $\forall i \in I_1, \tilde{x}_i = 1$, and: $\forall i \in I_0 \setminus \{k\}, \tilde{x}_i = 0$, then: $\Delta_j(I_1') - \Delta_j(I_1) = D_j(I_1) - \big[D_j(I_1') + D_j(I_1)\big].\tilde{x}_k$.*

**Proof:** (i) and (ii) are clear since: $|I_1| - 1 - \sum_{i \in I_1} \tilde{x}_i + \sum_{i \in I_0 \setminus \{k\}} \tilde{x}_i \geq -1$. (iii) is obvious. □

Since $D_j \geq 0$ (cf Lemma 3), points (i) and (ii) of the above lemma underline that we have interest in taking $\tilde{x}_k$ "large enough". On the one hand, in case (i), if $\tilde{x}_k$ is "too small", $\Delta_j(I_1') \geq \Delta_j(I_1)$ and including $k$ in $I_1$ does not improve the cut. On the other hand, in case (ii), the larger is $\tilde{x}_k$, the deeper is the cut.

Based on these observations, we propose the following greedy construction of $I_1$:

| **Greedy construction of $I_1$** | |
|---|---|
| **Step 1:** | Sort values $\{\tilde{x}_i\}_{i\in I_u(j)}$ in non-increasing order. |
| | Let $\varphi(r)$ denote the index of the $r^{th}$ element in the sorted list. |
| **Step 2:** | Set $I_1 = \{i \in I_u(j) \| \tilde{x}_i = 1\}$ and $r = \|I_1\| + 1$. |
| **Step 3:** | Set $I_1' = I_1 \cup \{\varphi(r)\}$ and compute $D_j(I_1')$. |
| **Step 4:** | Compute $\Delta_j(I_1') - \Delta_j(I_1)$. |
| | If $\Delta_j(I_1') < \Delta_j(I_1)$, set $I_1 = I_1'$. |
| **Step 5:** | If $r = \|I_u(j)\|$, STOP. |
| | Set $r \leftarrow r + 1$ and go to Step 3. |

This algorithm builds $I_1$ iteratively by considering first indices corresponding to the largest values $\{\tilde{x}_i\}_{i\in I_u(j)}$. When an index improves the current set $I_1$, it is added. The greedy construction of $I_1$ makes that it contains all indices $i$ such that $\tilde{x}_i = 1$.

**Lemma 14** *Let $j \in J$. If $\tilde{x}_i \in \{0, 1\}$ for all $i \in I_u(j)$, then the greedy construction of $I_1$ leads to the most violated inequality (4).*

**Proof:** Recall that: $\Delta_j(I_1) = b_j - \underline{A}_j\tilde{x} + \big(|I_1| - 1 - \sum_{i\in I_1}\tilde{x}_i + \sum_{i\in I_0}\tilde{x}_i\big).D_j(I_1)$. The most violated inequality is obtained with $I_1 = \{i \in I_u(j)|\tilde{x}_i = 1\}$ (see e.g. the proof of Lemma 5). This is the initial set built at Step 2. Let us show that this set will not be extended afterwards. Let any $k \in I_0$, $\tilde{x}_k = 0$ and: $\Delta_j(I_1') - \Delta_j(I_1) = D_j(I_1) \geq 0$, with $I_1' = I_1 \cup \{k\}$. Hence, the algorithm stops without extending the initial set $I_1$. □

This observation means that, even if the above separation process does not provide the most violated inequality in general, this property holds for integer points. As a consequence, when using this greedy algorithm in a branch-and-cut algorithm, we are sure to obtain finally an optimal solution. Indeed, infeasible integer solutions will be cut.

**Exact separation in a special case:** In this paragraph, let us consider the special case when the $\{\eta_{ji}\}_{i\in I_u(j)}$ are i.i.d.. Recall that under this assumption, $D_j(I_1)$ only depends on the cardinality $|I_1|$ (cf Lemma 4). Then, for each positive integer $k \leq |I_u(j)| = n_j$, solving the following problem is easy:

$$\min\left\{\Delta_j(I_1) \mid I_1 \subseteq I_u(j), |I_1| = k\right\}.$$

In this problem, since the cardinality of the set $I_1$ is fixed, $D_j(I_1)$ is a constant (cf Lemma 4). Then, the optimal solution $I_1^*$ consists of the $k$ indices corresponding to the largest values of $\{\tilde{x}_i\}_{i\in I_u(j)}$. Indeed, from equation (8): $\Delta_j(I_1) = b_j - \underline{A}_j.\tilde{x} + (k - 1 - \sum_{i\in I_1}\tilde{x}_i + \sum_{i\in I_0}\tilde{x}_i).D_j(I_1)$. Then, obtaining the most violated inequality is achieved by enumerating all the possible cardinalities, from $k = 1$ to $k = n_j$.

From a complexity viewpoint, this separation is done in time $O(n_j \log(n_j))$ (sorting complexity). Note that pre-computing the $n_j$ values of $D_j$ can be done once and for all at the beginning. Relative algorithms are described in the next section.

Finally, this special case validates the orientation taken in the greedy heuristic proposed for the general case. Indeed, the role played by the largest components of $\tilde{x}$ is well illustrated here.

## 5.2 Computation of $D_j(I_1)$

The separation process described above requires the computation of $D_j(I_1)$ (cf Step 4 of the greedy algorithm). Our aim is to achieve an efficient search by dichotomy, that requires a lower bound and an upper bound on this quantity. In Lemma 3, it has been shown that $D_j(I_1) > 0$. Now, some results are given to define an upper bound on $D_j(I_1)$.

Recall that we have defined a basic scenario satisfying the property (1); it can be seen as a "lower" scenario. Symmetrically, let us introduce an "upper" scenario, denoted by $\overline{A}$, and satisfying:

$$\forall I_1 \subseteq I_u(j), I_1 \neq \phi : \mathbb{P}\left(\sum_{i\in I_1}(A_{ji} - \overline{A}_{ji}) \leq 0\right) \geq 1 - \varepsilon_j. \tag{9}$$

As for the basic scenario, some rules can be derived for building such a scenario:

**Lemma 15** *Let $j \in J$, let us denote: $n_j = |I_u(j)|$. Suppose that $\overline{A}_j$ is defined as follows:*

*(i) when no particular probabilistic assumption is made, for all $i \in I_u(j)$: $\mathbb{P}(A_{ji} \leq \overline{A}_{ji}) \geq 1 - \varepsilon_j / n_j$;*

*(ii) if the r.v.s $\{A_{ji}\}_{i \in I_u(j)}$ are independent, for all $i \in I_u(j)$: $\mathbb{P}(A_{ji} \leq \overline{A}_{ji}) \geq (1 - \varepsilon_j)^{1/n_j}$.*

*Then $\overline{A}_j$ satisfies (9).*

**Proof:** Let $I_1 \subseteq I_u(j)$. We have:

$$\begin{aligned}\mathbb{P}\Big(\textstyle\sum_{i \in I_1} (A_{ji} - \overline{A}_{ji}) \leq 0\Big) &\geq \mathbb{P}\Big(\forall i \in I_u(j), A_{ji} \leq \overline{A}_{ji}\Big) \\ &= 1 - \mathbb{P}\Big(\exists i \in I_u(j) \text{ s.t. } A_{ji} > \overline{A}_{ji}\Big).\end{aligned}$$

When no particular assumption is made, we always have: $\mathbb{P}(\exists i \in I_u(j) \text{ s.t. } A_{ji} > \overline{A}_{ji}) \leq \sum_{i \in I_u(j)} \mathbb{P}(A_{ji} > \overline{A}_{ji})$. Assuming (i), this ensures: $\mathbb{P}\big(\sum_{i \in I_1} (A_{ji} - \overline{A}_{ji}) \leq 0\big) \geq 1 - n_j.\varepsilon_j / n_j = 1 - \varepsilon_j$.

Suppose now that r.v.s are independent. Then: $\mathbb{P}\big(\forall i \in I_u(j), A_{ji} \leq \overline{A}_{ji}\big) = \prod_{i \in I_u(j)} \mathbb{P}(A_{ji} \leq \overline{A}_{ji})$. From (ii), we deduce: $\mathbb{P}\big(\sum_{i \in I_1} (A_{ji} - \overline{A}_{ji}) \leq 0\big) \geq 1 - \varepsilon_j$. □

Now, the following result provides the required upper bound for $D_j(I_1)$:

**Lemma 16** *Let $j \in J$ and $I_1 \subseteq I_u(j)$ non-empty: $D_j(I_1) \leq \sum_{i \in I_1} (\overline{A}_{ji} - \underline{A}_{ji})$.*

**Proof:** Observe that: $\mathbb{P}\Big(\sum_{i \in I_1} \eta_{ji} \leq \sum_{i \in I_1} (\overline{A}_{ji} - \underline{A}_{ji})\Big) = \mathbb{P}\Big(\sum_{i \in I_1} A_{ji} \leq \sum_{i \in I_1} \overline{A}_{ji}\Big) \geq 1 - \varepsilon_j$, by definition of $\overline{A}_{ji}$. This proves that $D_j(I_1) \leq \sum_{i \in I_1} (\overline{A}_{ji} - \underline{A}_{ji})$. □

Then, $D_j(I_1)$ can now be computed by dichotomy by the following algorithm:

| **Algorithm for computing** $D_j(I_1)$ | |
|---|---|
| **Step 0:** | Set $D_{min} = 0$ and $D_{max} = \sum_{i \in I_1} (\overline{A}_{ji} - \underline{A}_{ji})$.<br>Let $\delta > 0$ be the precision wanted on the result. |
| **Step 1:** | If $D_{max} - D_{min} < \delta$, set $D = D_{min}$ and STOP. |
| **Step 2:** | Set $D = (D_{min} + D_{max})/2$, and compute the probability:<br>$\Psi = \mathbb{P}\left(\sum_{i \in I_1} \eta_{ji} \leq D\right)$ |
| **Step 3:** | If $\Psi < 1 - \varepsilon_j$, set $D_{min} = D$ and go to Step 1.<br>Otherwise, set $D_{max} = D$ and go to Step 1. |

The final quantity $D$ corresponds to the required protection term, with precision $\delta > 0$.

**Pre-computations of protection terms $D_j(I_1)$:** It has been said that when the $\{\eta_{ji}\}_{i \in I_u(j)}$ are i.i.d., the protection terms $D_j(I_1)$ only depend on the cardinality $n_1 = |I_1|$ (cf Lemma 4). As a consequence, these coefficients are in small number and can be easily pre-computed. This observation can be extended to the case when the $\{\eta_{ji}\}_{i \in I_u(j)}$ are independent, and distributed according to a small number of different distributions. This intuition is detailed further in Appendix B.

## 5.3 Computation of probabilities

Probability computation is a key algorithmic part. It is needed to test when a solution is feasible or not, i.e. to test if $\mathbb{P}(A_j x \leq b_j) \geq 1 - \varepsilon_j$ (for $j \in J$). It is needed also when computing terms $D_j(I_1)$: the algorithm of Section 5.2 shows that we have to be able to assess $\mathbb{P}\left(\sum_{i \in I_1} \eta_{ji} \leq D\right)$ for some $D$ (cf algorithm for computing $D_j(I_1)$).

Several approaches may be thought of for computing probabilities.

**Analytical formulas:** $D_j(I_1)$ is defined from a sum of r.v.s., namely: $\sum_{i \in I_1} \eta_{ji}$. In many specific cases, the distribution of such a sum is known analytically. This is the case, for instance, when the

$\{\eta_{ji}\}_{i\in I_u(j)}$ are independent and uniformly distributed on [0,1]. Then, denoting $n_1 = |I_1|$, classical probability results lead to (see e.g. [12]):

$$\begin{aligned}\mathbb{P}\left(\sum_{i=1}^{n_1}\eta_{ji}\le d\right) &= \frac{1}{n_1!}\sum_{k=0}^{\lfloor d\rfloor-1}\sum_{l=0}^{k}(-1)^l\binom{n_1}{l}\Big[(k+1-l)^{n_1}-(k-l)^{n_1}\Big]\\ &\quad+\frac{1}{n_1!}\sum_{l=0}^{\lfloor d\rfloor}(-1)^l\binom{n_1}{l}\Big[(d-l)^{n_1}-(\lfloor d\rfloor-l)^{n_1}\Big].\end{aligned} \tag{10}$$

This result can be used as a basis to obtain exact characterizations of various other probability distributions. Obviously, it is readily usable for independent r.v.s uniformly distributed on any interval $[l,u]$, $l<u$. Furthermore, any distribution which can be obtained from a sum of uniformly distributed r.v.s can be described easily from (10). For illustration, suppose that the $\{\eta_{ji}\}_{i\in I_u(j)}$ are independent and have symmetrical triangular distributions on [0,2]. Observe that each $\eta_{ji}$ can in fact be represented as the sum of two independent r.v.s uniformly distributed on [0,1]. Then, the cumulative distribution of $\sum_{i\in I_1}\eta_{ji}$ is obtained from (10) by replacing $n_1$ with $2n_1$.

Other analytical results may exist on sums of r.v.s, and may be used in our context. For instance, we refer to [7] for the sum of independent non-identical uniform r.v.s. Note that these exact formulas are very difficult to compute numerically when $n_1$ is large. When reaching such numerical problems, the recourse to the central limit theorem may provide very accurate approximations. Indeed, this very classical result ensures that, if the $\{\eta_{ji}\}_{i\in I_u(j)}$ are i.i.d.:

$$\lim_{n_1\to\infty}\mathbb{P}\left(\frac{\sum_{i=1}^{n_1}\eta_{ji}-n_1\mu}{\sigma\sqrt{n_1}}\le d\right)=\Phi(d)$$

where $\mu$ and $\sigma$ are respectively the expected value and the standard deviation of $\eta_{ji}$, and $\Phi$ denotes the cumulative distribution of the standard normal distribution $\mathcal{N}(0,1)$. In practice, this means that the cumulative distribution of $\mathcal{N}(n_1\mu, n_1\sigma^2)$ provides an approximation of the one of $\sum_{i=1}^{n_1}\eta_{ji}$. Note that extensions of the central limit theorem exist when the $\{\eta_{ji}\}_{i\in I_u(j)}$ are not identically distributed (see appendix C).

**Sampling techniques:** Monte-Carlo type algorithms can always be used:

**Lemma 17** *Let $q\in\mathbb{N}$ and $\varepsilon\in(0,1)$. Let $S$ be an r.v. taking values in $\mathbb{R}^q$, and let $t\in\mathbb{R}^q$. We denote: $\Psi=\mathbb{P}(S\le t)$. Let $\mu_1,\dots,\mu_k$ be a random sample from density $\mathbb{1}_{\{S\le t\}}$, we denote: $\bar\mu=\sum_{p=1}^k\mu_p/k$. Furthermore, let us introduce $\delta=|\bar\mu-1+\varepsilon|$:*

- *if $\Psi>1-\varepsilon$, we have:* $\Pr\Big(\bar\mu<1-\varepsilon\Big)\le\dfrac{1}{4k\delta^2}$;
- *if $\Psi<1-\varepsilon$, we have:* $\Pr\Big(\bar\mu>1-\varepsilon\Big)\le\dfrac{1}{4k\delta^2}$.

**Proof:** Let us denote: $T=\mathbb{1}_{\{S:S\le t\}}$, we have: $\Psi=\mathbb{E}[T]$. Suppose that $\Psi>1-\varepsilon$, then: $\Pr\Big(\bar\mu<1-\varepsilon\Big)\le\Pr\Big(\mid\bar\mu-\Psi\mid>\delta\Big)$. Then, from the weak law of large numbers: $\Pr\Big(\bar\mu<1-\varepsilon\Big)\le\sigma^2/(k\delta^2)$, where $\sigma^2$ is the variance of $T$. Furthermore, we have: $\sigma^2=\mathbb{E}[T^2]-\mathbb{E}[T]^2=\mathbb{E}[T^2]-\Psi^2$. Since $T$ takes values in $\{0,1\}$, we obtain: $\sigma^2=\mathbb{E}[T]-\Psi^2=\Psi(1-\Psi)\le1/4$.

The case when $\Psi<1-\varepsilon$ is exactly similar. □

This principle can be used each time a probability has to be assessed. For instance, we could evaluate the feasibility of row $j$ for a given $\bar x\in\{0,1\}^n$ by taking: $q=1$, $\varepsilon=\varepsilon_j$, $S=A_j\bar x$, and $t=b_j$. Lemma 17 makes it possible to decide whether $\mathbb{P}(A_jx\le b_j)\ge1-\varepsilon_j$ or not with controlled guarantees.

As a consequence, the resulting solution algorithm for CCILP is randomized. The error probability on the feasibility and optimality of an obtained solution can be computed easily. The advantage of sampling-based methods is that they can fit easily very complex probabilistic models and assumptions (various distributions, correlations, etc). However, their main drawback is the large number of simulations needed if we want to ensure good theoretical guarantees.

## 5.4 The case of joint probabilistic constraints

As stated in Section 2.2, $X$ is a relaxation of $X'$ when $\varepsilon_j = \varepsilon$ for all $j \in J$. Hence, the branch-and-bound algorithm described before can be used to solve: $\max\{cx \mid \mathbb{P}(Ax \leq b) \geq 1-\varepsilon\}$. We just have to discard the points in $X \setminus X'$. More precisely, when obtaining an integral solution $\bar{x}$ at a given node of the branching tree, we have to check whether it is actually a feasible solution of the chance-constrained problem or not. This is performed by computing the probability $\mathbb{P}(A\bar{x} \leq b)$: if it is larger than $1-\varepsilon$, we have indeed a feasible solution for CCILP (i.e. $\bar{x} \in X'$). Otherwise, the solution $\bar{x}$ should be discarded from the feasibility set. This is easily done just by ignoring this solution in the branching tree. Alternatively, the solution can be separated from the feasible polyhedron by adding an inequality: clearly, a 0-1 solution $x$ is different from $\bar{x}$ if, and only if: $\sum_{i\in I}(x_i - \bar{x}_i)^2 \geq 1$. Since all components are 0-1, this condition can be written equivalently: $\sum_{i\in I}(x_i + \bar{x}_i - 2\bar{x}_i x_i) \geq 1$. In practice, it is sufficient to add this latter linear inequality in our relaxation to ensure that $\bar{x}$ is removed from the set of feasible solutions.

## 5.5 Heuristic for CCILP

The solution algorithm may be accelerated if we know good lower bounds. Let us assume in this section that both $\underline{A}$ and $b$ are integral matrix and vector. A heuristic for computing feasible solutions to CCILP has been proposed in [17]:

**Heuristic for CCILP**

**Step 0:** Set $\alpha_j = 0$ for all $j \in J$.

**Step 1:** Let $x^*$ be an optimal solution of the problem:

$$\begin{aligned} \max \quad & cx \\ \text{s.t.} \quad & \underline{A}_j x \leq b_j - \alpha_j, \quad \forall j \in J, \\ & x \in \{0,1\}^n. \end{aligned}$$

**Step 2:** Test if $x^*$ is feasible for CCILP.

**Step 3:** If $x^*$ is not a feasible solution:

for all $j \in J$: set $\alpha_j \leftarrow b_j - \underline{A}_j.x^*$;

select $k$ such that: $k = \arg\min_{j\in J}\{b_j - \mathbb{E}[A_j x^*]\}$;

set $\alpha_k \leftarrow \alpha_k + 1$ and go to Step 1.

**Step 4:** If $x^*$ is a feasible solution, STOP.

The idea is very simple and natural: the right-hand-side terms of the problem are progressively decreased by increasing $\alpha_j$ (Step 3). Hence, the problem solved at Step 1 is more and more constrained. As a result, the feasibility probability of the computed solution tends to increase. We stop as soon as a solution for CCILP is found (Step 4).

# 6 Numerical tests

Relying on the separation process described in Section 5.1, we implement a branch-and-cut algorithm. Computer programs have been written in C++, using Cplex 10.0 for branch-and-cut algorithms. Note that all the automatic cut generations of Cplex have been turned off. All tests have been performed on a computer running an Intel(R) Xeon(TM) processor 2.8GHz with 2Go of RAM. The times indicated in tables are CPU times. All running times are limited to 900 seconds (15 minutes). In result tables, “t.l.” means that this time limit has been reached.

The presented experiments focus on cases when the r.v.s $\eta_{ji}$ are all i.i.d.. The reason is that this framework is the easiest to handle, and it may correspond to many real-life problems. However, recall that similar algorithms are usable in far more general cases.

### 6.1 Tests on multi-dimensional knapsack problems

Numerical tests are performed on probabilistic multi-dimensional knapsack problems:

$$\max \left\{ \sum_{i \in I} p_i x_i \mid \mathbb{P}\Big( \sum_{i \in I} w_{ji} x_i \le c_j \Big) \ge 1 - \varepsilon_j, \forall j \in J \right\}.$$

As before, $n = |I|$ and $m = |J|$. For all $j \in J$, $c_j > 0$ is the capacity of knapsack $j$. For each $i \in I$, $p_i > 0$ is the profit of taking element $i$ in the knapsack. $w_{ji}$ is a r.v. uniformly distributed on an interval $[\underline{w}_{ji}, \overline{w}_{ji}]$, with $\underline{w}_{ji} > 0$ and: $\overline{w}_{ji} = \underline{w}_{ji} + \delta$, with $\delta > 0$. With this latter assumption, the r.v.s $\{\eta_{ji}\}_{(i,j) \in I \times J}$ will be i.i.d.. In this setting, it has been shown in Section 5.1 that the separation of inequalities (4) can be done very efficiently.

We make tests with $\varepsilon_j = 0.1$ for all $j \in J$. This means that we look for the best solution satisfying each knapsack constraint with probability at least 0.9. Data $p_i$ and $\underline{w}_{ji}$ are randomly chosen in $[100, 1000]$, and $\delta = 20$. $c_j$ is randomly chosen in the interval $\big[1/3. \sum_{i \in I} \underline{w}_{ji}, 2/3. \sum_{i \in I} \underline{w}_{ji}\big]$.

As explained in Section 4, the basic scenario chosen may have an impact on the efficiency of the proposed solution algorithm. Three different basic scenarios are tested:

- B1: the first one is composed with the lowest weight values $\{\underline{w}_{ji}\}_{(i,j) \in I \times J}$;
- B2: using point (iii) of Lemma 8, the second basic scenario tested is built from average values $\{(\underline{w}_{ji} + \overline{w}_{ji})/2\}_{(i,j) \in I \times J}$;
- B3: finally, an improved basic scenario is built from B2 in the direction $v = (1, \ldots, 1)$, using (7) (cf Lemma 10). Note that, from Lemma 12, the obtained basic scenario is non-dominated, since the $\{\eta_{ji}\}_{(i,j) \in I \times J}$ are i.i.d..

Protection terms $D_j(I_1)$ are pre-computed, since they only depend on cardinality $|I_1|$ (cf Lemma 4). We rely on formula (10). However, numerical instabilities occur for $n_1 = |I_1| \ge 70$. To palliate this difficulty, inspired by the central limit theorem, we used the gaussian distribution $\mathcal{N}(n_1 \delta/2, n_1 \delta^2/12)$ (cf Section 5.3). An experimental comparison illustrates the convergence of this gaussian approximation to the exact value (see Figure 1). The absolute error of the gaussian approximation appears to be small (less than 0.1 when $n_1 \ge 50$). The relative error converges very fast to zero. Hence, we used this gaussian formula instead of the exact analytical one when $n_1 \ge 50$. The numerical instabilities encountered are illustrated on Figure 2.

Several cut generation strategies have been tested. We observed that generating aggressively violated inequalities lead to longer run times, even though the number of computation nodes was significantly reduced. Then, for each branching node, it appeared more efficient to generate at most one violated inequality per constraint row. The only exception is the root node, at which cuts are generated until no more violated one is found. This strategy is applied in all the following tests.

Results are reported in tables 1-5 for problems with $n = 200$ items, and $m = 1$ or $m = 5$ dimensions. With the naive approach B1, none of the test instances are solved to optimality within the time limit imposed (see Table 1). The proved gap to optimum is quite large on average (1.46%), and in fact an optimal solution has not been found in most of the cases (compare with Table 3, where all instances are solved to optimality). Using basic scenarios B2, for $m = 1$, the algorithm is much improved (see Table 2). 7 of the 10 instances are solved to optimality in less than 2 minutes. However, 3 of the instances are not completely solved within time limit, even though the proved gap is small. Finally, the comparison with Table 5 shows the great improvement brought from using B3 instead of B2. The maximal solution time is now less than 3 seconds, all instances being solved to optimality. When $m = 5$, similar observations can be derived from Tables 4 and 5. The solution time with basic scenarios B3 is kept very reasonable (up to 80 seconds).

When working with the basic scenario B3, the impact of the added inequalities is investigated by looking at the gap closure at the root node of the branching tree, for $m = 1$. The gap is measured between the initial linear relaxation, containing only the basic inequality $\underline{w}x \le c$, and a heuristic solution obtained from the algorithm of Section 5.5. Results appear in Table 6. As expected, the impact of cutting inequalities is moderate, since the gap closure is of only 5.72%.

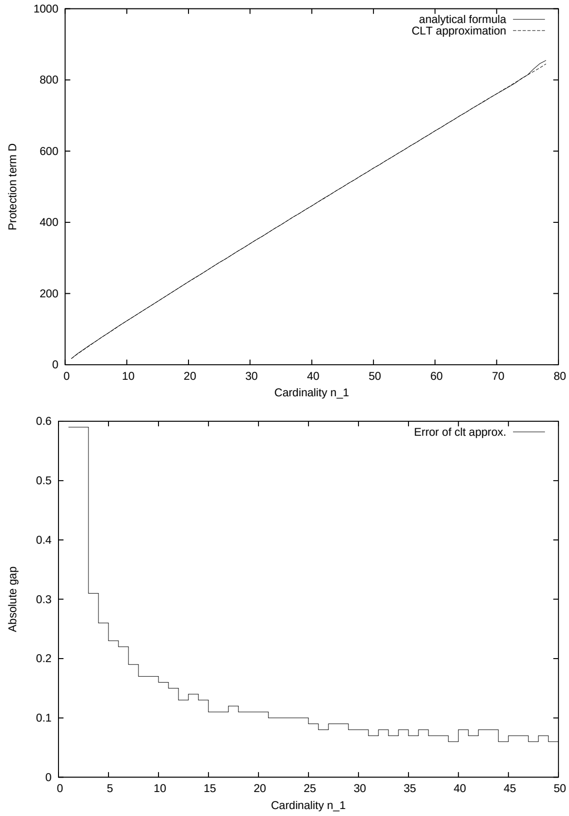



Figure 1: Uniform distributions: comparison of the exact computation of terms $D_j(I_1)$ with the approximation derived from the central limit theorem (CLT), in function of $n_1 = |I_1|$.

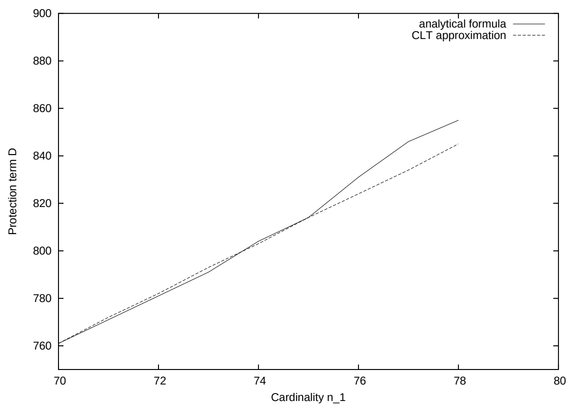



Figure 2: Uniform distributions: illustration of numerical instabilities resulting from the use of the analytical formula.

Table 1: Results for 10 knapsack problems with $n = 200$ ($m = 1$), with basic scenarios B1.

| instance | nb of nodes | nb of inequalities | obj. value | gap | B&C time (sec.) |
|---|---|---|---|---|---|
| 0 | 131398 | 35719 | 89190 | 1.25% | t.l. |
| 1 | 110874 | 29329 | 86879 | 1.22% | t.l. |
| 2 | 103431 | 27887 | 89336 | 1.23% | t.l. |
| 3 | 86143 | 23047 | 75059 | 1.59% | t.l. |
| 4 | 83101 | 22596 | 73426 | 1.38% | t.l. |
| 5 | 88401 | 22786 | 99069 | 1.29% | t.l. |
| 6 | 84831 | 23108 | 92005 | 1.71% | t.l. |
| 7 | 25431 | 8088 | 64992 | 2.17% | t.l. |
| 8 | 68401 | 20507 | 70874 | 1.44% | t.l. |
| 9 | 96051 | 24762 | 87241 | 1.29% | t.l. |
| average | 87806.2 | 23782.9 | 82807.1 | 1.46% | t.l. |

Table 2: Results for 10 knapsack problems with $n = 200$ ($m = 1$), with basic scenarios B2.

| instance | nb of nodes | nb of inequalities | optimum | gap | B&C time (sec.) |
|---|---|---|---|---|---|
| 0 | 1521 | 125 | 89345 | 0 | 5.85 |
| 1 | 26228 | 1767 | 86879 | 0 | 91.5 |
| 2 | 6224 | 419 | 89383 | 0 | 22.75 |
| 3 | 22361 | 8816 | 75232 | 0.06% | t.l. |
| 4 | 22700 | 8791 | 73446 | 0.07% | t.l. |
| 5 | 10339 | 546 | 99313 | 0 | 40.18 |
| 6 | 610 | 74 | 92647 | 0 | 3.2 |
| 7 | 5592 | 2123 | 65483 | 0 | 75.71 |
| 8 | 20663 | 8643 | 70880 | 0.08% | t.l. |
| 9 | 1189 | 117 | 87333 | 0 | 6.98 |
| average | 11742.7 | 3142.1 | 82994.1 | 0.02% | 307.14 |

Table 3: Results for 10 knapsack problems with $n = 200$ ($m = 1$), with basic scenarios B3.

| instance | nb of nodes | nb of inequalities | optimum | B&C time (sec.) |
|---|---|---|---|---|
| 0 | 202 | 8 | 89345 | 0.81 |
| 1 | 781 | 22 | 86879 | 2.56 |
| 2 | 274 | 12 | 89383 | 1.03 |
| 3 | 776 | 29 | 75254 | 2.79 |
| 4 | 207 | 4 | 73485 | 0.61 |
| 5 | 176 | 15 | 99313 | 0.74 |
| 6 | 24 | 0 | 92647 | 0.09 |
| 7 | 455 | 11 | 65483 | 1.89 |
| 8 | 363 | 28 | 70916 | 1.38 |
| 9 | 42 | 1 | 87333 | 0.17 |
| average | 330 | 13 | 83003.8 | 1.21 |

Table 4: Results for 10 multi-dimensional knapsack problems with $n = 200$ and $m = 5$, with basic scenarios B2.

| instance | nb of nodes | nb of inequalities | optimum | gap | B&C time (sec.) |
|---|---|---|---|---|---|
| 0 | 27661 | 4784 | 81353 | 0.08% | t.l. |
| 1 | 13094 | 9776 | 63987 | 0.11% | t.l. |
| 2 | 11301 | 8381 | 61986 | 0.20% | t.l. |
| 3 | 12369 | 8965 | 67729 | 0.05% | t.l. |
| 4 | 14039 | 6360 | 64082 | 0.12% | t.l. |
| 5 | 11497 | 8580 | 70711 | 0.12% | t.l. |
| 6 | 12301 | 7186 | 61663 | 0.16% | t.l. |
| 7 | 11301 | 6741 | 68490 | 0.15% | t.l. |
| 8 | 10667 | 8640 | 74675 | 0.11% | t.l. |
| 9 | 11198 | 9550 | 62930 | 0.07% | t.l. |
| average | 13542.8 | 7896.3 | 67760.6 | 0.12% | t.l. |

Table 5: Results for 10 multi-dimensional knapsack problems with $n = 200$ and $m = 5$, with basic scenarios B3.

| instance | nb of nodes | nb of inequalities | optimum | B&C time (sec.) |
|---|---|---|---|---|
| 0 | 3349 | 14 | 81353 | 41.03 |
| 1 | 1020 | 0 | 64022 | 12.87 |
| 2 | 6296 | 42 | 62018 | 80.53 |
| 3 | 299 | 1 | 67745 | 3.49 |
| 4 | 329 | 8 | 64127 | 5.28 |
| 5 | 2719 | 18 | 70731 | 37.07 |
| 6 | 1266 | 24 | 61708 | 17.38 |
| 7 | 3250 | 27 | 68520 | 41.07 |
| 8 | 1046 | 0 | 74706 | 12.57 |
| 9 | 1333 | 11 | 62930 | 16.74 |
| average | 2090.7 | 14.5 | 67786 | 26.8 |

Table 6: Impact of cutting inequalities at root node, when $n = 200$ and $m = 1$.

| instance | nb of ineq. | heuristic value | init. relaxation value | final relaxation value | gap closure |
|---|---|---|---|---|---|
| 0 | 2 | 89345 | 89400.2 | 89395.2 | 9.14% |
| 1 | 2 | 86879 | 86926.6 | 86925 | 3.43% |
| 2 | 2 | 89383 | 89431.6 | 89428.6 | 6.29% |
| 3 | 2 | 75254 | 75330.4 | 75326.1 | 5.55% |
| 4 | 2 | 73485 | 73528.7 | 73527.6 | 2.36% |
| 5 | 2 | 99313 | 99343.9 | 99341.6 | 7.27% |
| 6 | 2 | 92647 | 92681.6 | 92678.4 | 9.19% |
| 7 | 2 | 65483 | 65564.9 | 65558.9 | 7.23% |
| 8 | 2 | 70916 | 70958.9 | 70957.5 | 3.25% |
| 9 | 2 | 87333 | 87362.9 | 87361.5 | 4.62% |
| average | 2 | 81726.75 | 81775.36 | 81772.53 | 5.72% |

## 6.2 The probabilistic capacitated facility location problem

Let us introduce a chance-constrained version of the very classical capacitated facility location problem. This problem has many applications in various fields. Let $I$ be the set of clients, and $J$ the set of possible locations for facilities. Each client $i \in I$ has a demand $d_i$, and the capacity of a facility is denoted by $C$. We suppose that each client has to be assigned to only one facility site. The transportation cost between a site $j$ and a client $i$ is denoted by $a_{ij}$, while $b_j$ denotes the cost of opening a facility at site $j$.

Each demand $d_i$ is assumed to be uncertain: it denotes a random variable. Then, the Probabilistic Capacitated Facility Location Problem (PCFLP) is:

$$\begin{array}{lll}\min & \sum_{i\in I}\sum_{j\in J} a_{ij}x_{ij} + \sum_{j\in J} b_j y_j & \\ \text{s.t.} & \sum_{j\in J} x_{ij} = 1, & \forall i \in I, \\ & \mathbb{P}\left(\sum_{i\in I} d_i x_{ij} \le C y_j\right) \ge 1-\varepsilon, & \forall j \in J, \\ & x_{ij} \in \{0,1\}, y_j \in \{0,1\}, & \forall i \in I, j \in J.\end{array}$$

$\varepsilon \in [0,1)$ is the unfeasibility tolerance for each site $j \in J$. It means that we accept not to be able to satisfy all the demand assigned to a site, but with controlled probability.

The implementation relies on the improved basic scenario obtained thanks to Lemma 10, in the direction $v = (1, \dots, 1)$.

**Triangular distributions:** We specify now the probabilistic assumptions. The $\{d_i\}_{i\in I}$ are assumed to be independent r.v.s. For each $i \in I$, $d_i$ has a symmetrical triangular distribution on an interval $[\underline{d}_i, \overline{d}_i]$ ($\underline{d}_i \ge 0$). More precisely, the distribution function is defined by:

$$f_i(x) = \begin{cases} 0, & \text{if } x \le \underline{d}_i \text{ or } x \ge \overline{d}_i, \\ \alpha_i.(x - \underline{d}_i), & \text{if } \underline{d}_i \le x \le (\underline{d}_i + \overline{d}_i)/2, \\ \alpha_i.(\overline{d}_i - x), & \text{if } (\underline{d}_i + \overline{d}_i)/2 \le x \le \overline{d}_i. \end{cases}$$

with: $\alpha_i = 2/(\overline{d}_i - \underline{d}_i)^2$.

Data are randomly generated: each minimal demand $\underline{d}_i$ is chosen integral between 50 and 500, each link cost between 10 and 100, each site cost between 200 and 300. The uncertainty on demands is fixed to: $\overline{d}_i - \underline{d}_i = 30$, for all $i \in I$. Finally, we consider $\varepsilon = 0.1$. The facility capacity is defined as: $C = \lceil 1.5 * \sum_{i\in I} \overline{d}_i / m \rceil$.

The use of gaussian assumptions for chance constrained problems is often motivated in the literature by the central limit theorem. The gaussian approximation model is built as described in Section 3.1. Namely, after simplifications, the problem obtained is:

$$\begin{array}{lll}\min & \sum_{i\in I}\sum_{j\in J} a_{ij}x_{ij} + \sum_{j\in J} b_j y_j & \\ \text{s.t.} & \sum_{j\in J} x_{ij} = 1, & \forall i \in I, \\ & \sum_{i\in I} \mathbb{E}[d_i] x_{ij} \le C y_j, & \forall j \in J, \\ & \sum_{i\in I}\left(\Phi^{-1}(1-\varepsilon)^2\sigma^2 - \mathbb{E}[d_i]^2\right).x_i + 2\sum_{i\in I}\mathbb{E}[d_i]C.v_{ji} & \\ & \quad -2\sum_{i<k}\mathbb{E}[d_i]\mathbb{E}[d_k].u_{jik} \le C^2 y_j, & \forall j \in J, \\ & u_{jik} \le x_{ij} \text{ and } u_{jik} \le x_{kj}, & \forall j \in J, (i,k) \in I^2, i<k, \\ & v_{ji} \ge y_j + x_{ij} - 1, & \forall j \in J, i \in I, \\ & x_{ij} \in \{0,1\}, y_j \in \{0,1\}, & \forall i \in I, j \in J, \\ & u, v \ge 0. & \end{array} \quad (11)$$

where $\sigma^2$ is the variance of one demand. With our assumptions: $\sigma^2 = 1/24$.

Results are reported in Table 7. Our algorithm performs quite well: optimal solutions are found for instances up to 50×15. When optimality is not reached, feasible solution are provided, with moderate proved gap. The instance sizes remain small enough to ensure that our so-called "exact algorithm" is indeed accurate: the probability assessments are all based on the analytical formula (and not on gaussian approximations). Then, our computational results show that the gaussian approximation model is very close to the exact model, as expected. However, this approximation appears far less tractable than our approach: solution times are a lot longer. Using this model, the problem size makes it even hard to solve the linear relaxation at the root node: it has not been solved within the imposed time limit for instances 100×20 and 100×30.

Table 7: Triangular distributions: results for the probabilistic capacitated facility location problem (PCFLP).

| Instance size | | Exact algorithm | | | | Gaussian deterministic equivalent | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | $m$ | nb of nodes | nb of ineq. | time (sec.) | value | nb of nodes | time (sec.) | value |
| 20 | 10 | 22 | 26 | 0.19 | 2152 | 14288 | t.l. (2.89%) | 2162 |
| 30 | 10 | 3 | 18 | 0.23 | 2369 | 2419 | 504.23 | 2369 |
| 50 | 10 | 72 | 54 | 1.41 | 2555 | 442 | 398.5 | 2555 |
| 50 | 15 | 246 | 111 | 6.72 | 3112 | 500 | t.l. (1.26%) | 3112 |
| 50 | 20 | 13872 | 4130 | t.l. (5.59%) | 3956 | 209 | t.l. (8.05%) | 3965 |
| 100 | 20 | 5772 | 1810 | t.l. (1.33%) | 5005 | no solution found within time limit | | |
| 100 | 30 | 2819 | 1591 | t.l. (3.86%) | 6422 | no solution found within time limit | | |

Table 8: Bernoulli distributions: results for the probabilistic capacitated facility location problem (PCFLP).

| Instance size | | Exact algorithm | | | | Gaussian deterministic equivalent | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | $m$ | nb of nodes | nb of ineq. | time (sec.) | value | nb of nodes | time (sec.) | value |
| 20 | 10 | 172 | 54 | 0.68 | 2144 | 14345 | t.l. (1.49%) | 2152 |
| 30 | 10 | 11 | 21 | 0.22 | 2368 | 811 | 175.11 | 2369 |
| 50 | 10 | 56 | 42 | 0.95 | 2551 | 404 | 358.36 | 2555 |
| 50 | 15 | 6985 | 1069 | 171 | 3108 | 460 | t.l. (3.60%) | 3112 |
| 50 | 20 | 14501 | 4214 | t.l. (5.64%) | 3952 | 86 | t.l. (9.95%) | 4050 |

**Bernoulli distributions:** Finally, let us suppose that each demand $d_i$ can take only two values $\underline{d}_i$ and $\overline{d}_i$, with respective probabilities $1-p$ and $p$, $p \in (0,1)$. Furthermore, let us assume that the gap between the upper and lower values is the same for all demands: for all $i \in I$, $\overline{d}_i - \underline{d}_i = \delta > 0$. Let us also denote $\eta_i = d_i - \underline{d}_i$. It is known that, for all $I_1 \subseteq I$ ($n_1 = |I_1|$), $\sum_{i \in I_1} \eta_i$ follows a binomial distribution:

$$\forall k \in \{0, \ldots, n_1\}, \mathbb{P}\left(\sum_{i \in I_1} \eta_i = k\delta\right) = \binom{n_1}{k} p^k (1-p)^{n_1-k}.$$

Note that, as soon as $p < 1-\varepsilon$, the scenario $\underline{d}$ satisfies the property (1): it is a basic scenario. In our specific implementation, all the data $\underline{d}_i$, $\overline{d}_i$ and $C$ have been generated as in the previous paragraph. As a result, $\delta = 30$. Furthermore, we take $p = 0.3$. (Note that the associated variance is $\sigma^2 = \delta^2 p(1-p)$.)

As previously, the gaussian approximation has been tested here, cf model (11). However, Figure 3 illustrates the difficulty in approximating a discrete distribution with a continuous one. Results are reported in Table 8. Our algorithm performs reasonably well: the performance is similar to this observed before with triangular distributions. Medium size instances are solved. The gaussian model is still a lot more difficult to solve. Furthermore, it provides solutions which are not always optimal for the initial problem considered (see instances 30×10 and 50×10). This is an additional motivation for solving chance-constrained problems with taking accurately account of specific probability assumptions instead of gaussian approximations.

# 7 Conclusion

In this paper, the optimal resolution of chance-constrained integer linear programs has been investigated. The set of feasible solutions has been characterized by an exponential number of linear inequalities. Then, the separation of these inequalities has been studied and relative algorithms have been designed. The whole analysis is based on finding a scenario, said basic, for which the investigated solution will be feasible. The basic scenario is shown to impact the quality of the linear relaxation defined. Theoretical

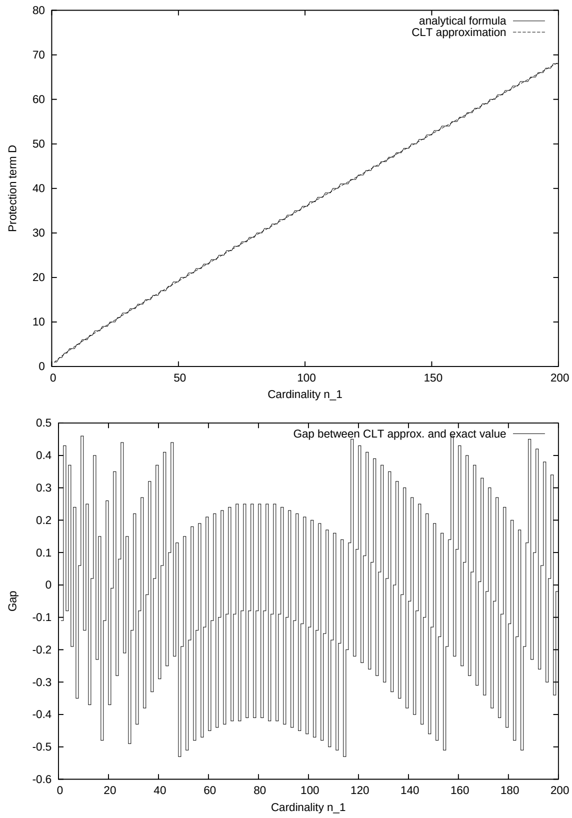



Figure 3: Bernoulli distributions: comparison of the exact computation of terms $D_j(I_1)$ with the approximation derived from the central limit theorem, in function of $n_1 = |I_1|$.

properties, as well as algorithms, are given to make it possible to obtain an efficient basic scenario. This appears to be of critical importance in practice in our algorithms. From a more general viewpoint, basic scenarios are not directly linked to our solution method. In fact, they will certainly be usable for improving other different solution approaches.

Numerical tests have been performed on multi-dimensional knapsack problems, as well as on capacitated location problems. They underline the impact of the different components of the solution method. Instances are solved to optimality up to medium sizes. Furthermore, it is classical in the literature to rely on gaussian probabilistic assumptions, from which equivalent deterministic models are easy to derive. We performed a comparison between using this classical equivalent deterministic model and our approach. It appeared that our framework is much more tractable in practice. Finally, the recourse to such specific gaussian assumptions is usually motivated by the central limit theorem. Our experiments show that such gaussian approximations remain imperfect, and may lead to solutions different from the real optimal one, obtainable with our algorithms.

# Appendix

## A Idea of generalization to non-linear programming

Consider the following more general set description:

$$X = \Big\{x \in \{0,1\}^n \mid \forall j \in J, \mathbb{P}\big(g_j(x,\xi) \leq 0\big) \geq 1 - \varepsilon_j\Big\}$$

$\xi$ is a random variable taking values in $\mathbb{R}^p$ ($p \in \mathbb{N}$). We assume that for each $j \in J$, there exists $\underline{\xi}_j \in \mathbb{R}^p$ such that, for all $x \in \{0,1\}^n$: either $\mathbb{P}\left(g_j(x,\xi) \leq g_j(x,\underline{\xi}_j)\right) < 1 - \varepsilon_j$, or $g_j(x,\underline{\xi}_j) \leq 0$.

Now, the main results exposed for the linear case in Section 3.2 can be transposed in this context.

**Lemma 18** *For all $j \in J$, the inequality $g_j(x,\underline{\xi}_j) \leq 0$ is valid for $X$.*

**Proof:** Let $x \in X$ and suppose that $g_j(x,\underline{\xi}_j) > 0$. Then: $\mathbb{P}\big(g_j(x,\xi) \leq 0\big) \leq \mathbb{P}\left(g_j(x,\xi) \leq g_j(x,\underline{\xi}_j)\right)$. Furthermore, from the definition of $\underline{\xi}_j$: $\mathbb{P}\left(g_j(x,\xi) \leq g_j(x,\underline{\xi}_j)\right) < 1 - \varepsilon_j$ (contradiction). □

For any $I_1 \subseteq I$, let us denote by $x(I_1) \in \{0,1\}^n$ the vector such that $x(I_1)_i = 1$ if and only if $i \in I_1$. We introduce:

$$D_j(I_1) = \sup\Big\{d \mid \mathbb{P}\Big(g_j(x(I_1),\xi) \leq g_j(x(I_1),\underline{\xi}_j) + d\Big) < 1 - \varepsilon_j\Big\}$$

From the assumptions made, $D_j(I_1) > 0$. Furthermore, as in Lemma 8, it is easy to show that: $\mathbb{P}\Big(g_j(x(I_1),\xi) \leq g_j(x(I_1),\underline{\xi}_j) + D_j(I_1)\Big) \geq 1 - \varepsilon_j$. The inequality:

$$g_j(x,\underline{\xi}_j) + D_j(I_1).\Big(\sum_{i \in I_1} x_i - \sum_{i \notin I_1} x_i\Big) \leq (|I_1| - 1)\,.D_j(I_1) \tag{12}$$

is valid for $X$. Indeed, when $x \neq x(I_1)$, the inequality is dominated by $g_j(x,\underline{\xi}_j) \leq 0$. When $x = x(I_1)$, we obtain: $g_j(x,\underline{\xi}_j) + D_j(I_1) \leq 0$, which can be proved to be valid, from our assumptions (cf the proof of Lemma 5).

**Proposition 2**

$$X = \Big\{x \in \{0,1\}^n \mid x \textit{ satisfies all inequalities (12)}\Big\}$$

**Proof:** We have already said that the inclusion $\subseteq$ was correct. Consider now $x \notin X$: there exists $j \in J$ such that $\mathbb{P}\big(g_j(x,\xi) \le 0\big) < 1 - \varepsilon_j$. Suppose that $x$ satisfies all inequalities (12), then in particular: $g_j(x,\underline{\xi}_j) + D_j(I_1) \le 0$, where $I_1 = \{i \in I | x_i = 1\}$. We deduce that: $\mathbb{P}\big(g_j(x,\xi) \le g_j(x,\underline{\xi}_j) + D_j(I_1)\big) \le \mathbb{P}\big(g_j(x,\xi) \le 0\big) < 1 - \varepsilon_j$. This is a contradiction with the definition of $D_j(I_1)$. $\square$

# B Pre-computation of coefficients $D_j(I_1)$

Suppose that the $\{\eta_{ji}\}_{i \in I_u(j)}$ are all independent, and that there exists a partition $\{U_1, \ldots, U_K\}$ of $I_u(j)$ such that, for all $k \in \{1, \ldots, K\}$, the $\{\eta_{ji}\}_{i \in U_k}$ are identically distributed. Let us denote by $N_K(c)$ the number of solutions $(u_1, \ldots, u_K)$ of the following problem:

$$\begin{cases} \sum_{k=1}^K u_k = c, \\ u_k \le |U_k|, \forall k \in \{1, \ldots, K\}, \\ u_k \in \mathbb{N}, \forall k \in \{1, \ldots, K\}. \end{cases}$$

Observe that:

- when $c > \sum_{k=1}^K |U_k|$ or $c < 0$: $N_K(c) = 0$;
- $N_K(0) = 1$ and $N_K(1) = K$;
- $N_K(|I_u(j)|) = 1$.

**Lemma 19** *For all $c \in \mathbb{Z}$: $N_1(c) = \mathbb{1}_{0 \le c \le |U_1|}$. When $K \ge 2$, for all $c \in \mathbb{Z}$:*

$$N_K(c) = \sum_{t=0}^{|U_K|} N_{K-1}(c-t).$$

**Proof:** It is sufficient to see that when we take $t$ elements in the set $U_K$ ($u_k = t$), then there are $N_{K-1}(c-t)$ different solutions possible. $\square$

Hence, $N_K(c)$ can be computed by induction. There is a total number of $\sum_{c=1}^{|I_u(j)|} N_K(c)$ different protection terms for the considered problem.

As an illustration, consider the case when $K = 2$. For all $c \in \mathbb{Z}$, $c \le |I_u(j)|$:

$$N_2(c) = \sum_{t=0}^{|U_2|} N_1(c-t) = \sum_{t=0}^{|U_2|} \mathbb{1}_{0 \le c-t \le |U_1|} = \sum_{t=(c-|U_1|)^+}^{\min\{c,|U_2|\}} 1 = \min\{c, |U_2|\} - (c - |U_1|)^+ + 1 \, .$$

Let us suppose that $|U_1| \le |U_2|$. Then, the total number of protection terms to be computed is:

$$\begin{aligned} \sum_{c=1}^{|I_u(j)|} N_2(c) &= \sum_{c=1}^{|U_1|} (c+1) + \sum_{c=|U_1|+1}^{|U_2|} (|U_1|+1) + \sum_{c=|U_2|+1}^{|I_u(j)|} (|I_u(j)| - c + 1) \\ &= |U_1|.(|U_1|+3)/2 + (|U_1|+1).(|U_2| - |U_1|) + |U_1|.(|U_1|+1)/2 \, . \end{aligned}$$

This number is represented on Figure 4 for $|I_u(j)| = 100$.

# C A generalized central limit theorem

It is widely known that the sum of r.v.s which are i.i.d. converges to a gaussian law (classical central limit theorem). Several extensions to this result exist, with weaker assumptions. We give the following one, taken from [21]:

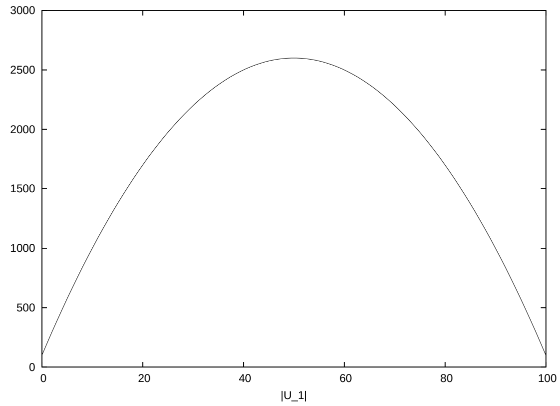



Figure 4: Number of protection terms to be pre-computed, when $|I_u(j)| = 100$ and $K = 2$.

**Theorem 1** *Let $(X_n)_{n\in I\!N}$ be a sequence of independent r.v.s. Let us suppose that each $X_n$ has finite expected value $\mu_n$ and variance $\sigma_n^2$. If:*

*(i) the r.v.s $\{X_n\}_{n\in I\!N}$ are uniformly bounded, i.e.: there exists a real $M$ such that: $\forall n \in I\!N, \mathbb{P}(|X_n| < M) = 1$;*

*(ii) $\sum_{n\geq 0} \sigma_n^2 = \infty$,*

*then $Z_n = \sum_{i=0}^{n}(X_i - \mu_i)/\sqrt{\sum_{i=0}^{n}\sigma_i^2}$ converges in law to the standard normal distribution:*

$$\forall z \in I\!R, \lim_{n\to\infty} \mathbb{P}(Z_n \leq z) = \Phi(z)$$

# References


[1] Aringheri, R.: A tabu search algorithm for solving chance-constrained programs. Note del polo 73, DTI - University of Milano (2005). Disponible à http://www.crema.unimi.it/Biblioteca/SchedaNota.asp?Nota=92

[2] Ben-Tal, A., Nemirovski, A.: Robust solutions of linear programming problems contamined with uncertain data. Mathematical Programming (Ser. A) **88**, 411–424 (2000)

[3] Beraldi, P., Ruszczyński, A.: A branch and bound method for stochastic integer problems under probabilistic constraints. Optimization Methods and Software **17**, 359–382 (2002)

[4] Bertsimas, D., Sim, M.: The price of robustness. Operations Research **52**(1), 35–53 (2004)

[5] Bianchi, L., Dorigo, M., Gambardella, L., Gutjahr, W.: Metaheuristics in stochastic combinatorial optimization: a survey. technical report idsia-08-06, IDSIA (2006). Disponible à www.idsia.ch/idsiareport/IDSIA-08-06.pdf

[6] Birge, J., Louveaux, F.: Introduction to Stochastic Programming. Springer-Verlag (1997)

[7] Bradley, D., Gupta, R.: On the distribution of the sum of non-identically distributed uniform random variables. Annals of the Institute of Statistical Mathematics **54**(3), 689–700 (2002)

[8] Calafiore, G., Campi, M.: Uncertain convex programs: randomized solutions and confidence levels. Mathematical Programming (Ser. A) **102**, 25–46 (2005)

[9] Calafiore, G., Ghaoui, L.E.: Distributionally robust chance-constrained linear programs with applications. Journal of Optimization Theory and Applications **130**(1), 1–22 (2006)

[10] Charnes, A., Cooper, W.: Chance-constrained programming. Management Science **5**, 73–79 (1959)

[11] Erdoğan, E., Iyengar, G.: Ambiguous chance constrained problems and robust optimization. Mathematical Programming (Ser. B) **107**(1-2), 37–61 (2006)

[12] Grinstead, C., Snell, J.: Introduction to Probability. American Mathematical Society (1997). (Second revised edition.)

[13] Haneveld, W.K., van der Vlerk, M.: Stochastic integer programming: general models and algorithms. Annals of Operations Research **85**, 39–57 (1999)

[14] Hillier, F.: Chance-constrained programming with 0-1 or bounded continuous decision variables. Management Science **14**(1), 34–57 (1967)

[15] Kall, P., Mayer, J.: Stochastic Linear Programming. Springer (2005)

[16] Kall, P., Wallace, S.: Stochastic Programming. Wiley, Chichester (1994)

[17] Klopfenstein, O.: Tractable algorithms for chance-constrained combinatorial problems (2007). Submitted.

[18] Klopfenstein, O., Nace, D.: A robust approach to the chance-constrained knapsack problem (2006). Submitted.

[19] Nemirovski, A., Shapiro, A.: Scenario approximations of chance constraints. In: G. Calafiore, F.D. (Eds.) (eds.) Probabilistic and Randomized Methods for Design under Uncertainty, pp. 3–48. Springer, London (2005)

[20] Nemirovski, A., Shapiro, A.: Convex approximations of chance constrained programs. SIAM Journal on Optimization **17**, 969–996 (2006)

[21] Ross, S.: Initiation aux probabilités. Presses Polytechniques Romandes (1987)

[22] Seppälä, Y., Orpana, T.: Experimental study on the efficiency and accuracy of a chance-constrained programming algorithm. European Journal of Operational Research **16**, 345–357 (1984)

[23] Tayur, S., Thomas, R., Natraj, N.: An algebraic geometry algorithm for scheduling in the presence of setups and correlated demands. Mathematical Programming **69**(3), 369–401 (1995)

[24] Weintraub, A., Vera, J.: A cutting plane approach for chance constrained linear programs. Operations Research **39**(5), 776–785 (1991)