Weierstraß-Institut
für Angewandte Analysis und Stochastik

im Forschungsverbund Berlin e.V.

Preprint ISSN 0946 – 8633

Discrepancy distances and scenario reduction in two-stage stochastic integer programming

René Henrion[1], Christian Küchler[2], Werner Römisch[2]

submitted: September 19, 2007

[1] Weierstrass Institute
for Applied Analysis
and Stochastics
Mohrenstr. 39
10117 Berlin
Germany
E-Mail: henrion@wias-berlin.de

[2] Humboldt University Berlin
Institute of Mathematics
10099 Berlin
Germany
E-Mail: ckuechler@math.hu-berlin.de
romisch@mathematik.hu-berlin.de

No. 1256
Berlin 2007

2000 *Mathematics Subject Classification.* 90C15.

Key words and phrases. Stochastic programming, two-stage, mixed-integer, chance constraints, scenario reduction, discrepancy, Kolmogorov metric.

This work was supported by the DFG Research Center MATHEON *Mathematics for key technologies* in Berlin, by the Wiener Wissenschafts-, Forschungs- und Technologiefonds (WWTF) and by the German Ministry of Education and Research (BMBF) under the grant 03SF0312E.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Mohrenstraße 39
10117 Berlin
Germany

Fax: + 49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract

Polyhedral discrepancies are relevant for the quantitative stability of mixed-integer two-stage and chance constrained stochastic programs. We study the problem of optimal scenario reduction for a discrete probability distribution with respect to certain polyhedral discrepancies and develop algorithms for determining the optimally reduced distribution approximately. Encouraging numerical experience for optimal scenario reduction is provided.

1 Introduction

Two-stage (linear) stochastic programs arise if, for given realizations ξ of a random vector and first-stage decisions x, possible violations of a constraint $Tx = h(\xi)$ are compensated by a second-stage decision $y(\xi)$, being nonnegative, satisfying $Wy(\xi) = h(\xi) - Tx$ with a (fixed) recourse matrix W, and inducing costs $\langle q(\xi), y(\xi)\rangle$. Then the idea is to minimize the sum of the expected recourse cost $\mathbb{E}(\langle q(\xi), y(\xi)\rangle)$ and of the first-stage cost $\langle c, x\rangle$ with x varying in a constraint set X. If some of the components of x and/or $y(\xi)$ are integer variables, one arrives at two-stage stochastic programs with mixed-integer linear recourse. Such optimization models may be rewritten in the form

$$\min\left\{\langle c, x\rangle + \int_{\Xi} \Phi(q(\xi), h(\xi) - Tx)\mathbb{P}(d\xi) : x \in X\right\}, \tag{1}$$

where Φ is a mapping from $\mathbb{R}^{m_1+m_2} \times \mathbb{R}^d$ to the extended reals given by

$$\Phi(u, t) := \min\{\langle u_1, y_1\rangle + \langle u_2, y_2\rangle : Wy_1 + \bar{W}y_2 = t, y_1 \in \mathbb{R}^{m_1}_+, y_2 \in \mathbb{Z}^{m_2}_+\}, \tag{2}$$

Thereby, $c \in \mathbb{R}^m$, X is a closed subset of $\mathbb{R}^m$, Ξ is a (convex) polyhedral subset of $\mathbb{R}^s$, $\mathbb{P}$ is a Borel probability measure on Ξ, W and $\bar{W}$ are (d, m_1)- and (d, m_2)-matrices, respectively, T is a (d, m)-matrix, and $q(\xi) \in \mathbb{R}^{m_1+m_2}$ and $h(\xi) \in \mathbb{R}^d$ are affine mappings of $\xi \in \mathbb{R}^s$.

The standard approach for solving mixed-integer two-stage stochastic programs of the form (1) consists in approximating the original probability distribution P by a discrete probability measure with N atoms ξ^i and probabilities p_i, $i = 1, \ldots, N$. The resulting stochastic program is equivalent to the mixed-integer program

$$\min\Big\{\langle c, x\rangle + \sum_{i=1}^{N} p_i(\langle q_1(\xi^i), y_{1i}\rangle + \langle q_2(\xi^i), y_{2i}\rangle : \tag{3}$$
$$x \in X, y_{1i} \in \mathbb{R}^{m_1}_+, y_{2i} \in \mathbb{Z}^{m_2}_+, Wy_{1i} + \bar{W}y_{2i} = h(\xi^i) - Tx, i = 1, \ldots, N\Big\},$$

where the number of integer variables y_{2i} increases with N. Since for applied stochastic optimization models (see, e.g., [17, 18]) often the dimension m_2 of each y_{2i} gets large, the mixed-integer program (3) might become huge even for small numbers N of scenarios and practically unsolvable for large N. Thus, in applications, it might be desirable or even inevitable to reduce the number of scenarios such that reasonable solution times for (3) are achieved.

Previous work [3, 7, 8] on scenario reduction for two-stage stochastic programs without integrality requirements suggests to base the reduction process on *suitable* distances of probability distributions. Roughly speaking, such a probability metric $\mathbb{D}$ is suitable if the optimal values $v(\cdot)$ and solution sets of the underlying stochastic program behave continuously at $\mathbb{P}$ if $\mathbb{P}$ is perturbed in terms of the distance $\mathbb{D}$. As argued in [22], (semi-)metrics with ζ-structure (cf. [20, 27]) of the form

$$\mathbb{D}_{\mathcal{F}}(\mathbb{P},\mathbb{Q}) := \sup_{f\in\mathcal{F}} \left| \int_{\Xi} f(\xi)\mathbb{P}(d\xi) - \int_{\Xi} f(\xi)\mathbb{Q}(d\xi) \right|, \tag{4}$$

with $\mathcal{F}$ denoting a certain class of Borel measurable functions from Ξ to $\mathbb{R}$, are suitable probability distances for many stochastic programs. By extending the results in [25, 22] it is shown in [23] that the class

$$\mathcal{F}_{r,\mathcal{B}}(\Xi) := \{f\mathbb{1}_B : f \in \mathcal{F}_r(\Xi), B \in \mathcal{B}\}$$

and the corresponding distance with ζ-structure

$$\zeta_{r,\mathcal{B}}(\mathbb{P},\mathbb{Q}) := \mathbb{D}_{\mathcal{F}_{r,\mathcal{B}}}(\mathbb{P},\mathbb{Q}) \tag{5}$$

between probability measures $\mathbb{P}$ and $\mathbb{Q}$ in $\mathcal{P}_r(\mathbb{R}^s)$ is suitable for (1) if $r = 2$ and $\mathcal{B}$ is a certain class of (convex) polyhedra in Ξ with a uniformly bounded number of faces. Here, $\mathbb{1}_B$ denotes the characteristic function of the set B, the class $\mathcal{F}_r(\Xi)$ consists of all continuous functions $f : \Xi \to \mathbb{R}$ such that the estimates

$$|f(\xi)| \le \max\{1, \|\xi\|^r\} \quad \text{and} \quad |f(\xi) - f(\tilde{\xi})| \le \max\{1, \|\xi\|^{r-1}, \|\tilde{\xi}\|^{r-1}\}\|\xi - \tilde{\xi}\|$$

hold true for all $\xi, \tilde{\xi} \in \Xi$, and $\mathcal{P}_r(\Xi)$ is the set of all Borel probability measures on Ξ having finite absolute moments of order $r \ge 1$. However, as pointed out in the Appendix, it seems to be difficult to employ the metrics $\zeta_{r,\mathcal{B}}$ for scenario reduction directly. An alternative to $\zeta_{r,\mathcal{B}}$ are the so-called $\mathcal{B}$-discrepancies (cf. [14])

$$\alpha_{\mathcal{B}}(\mathbb{P},\mathbb{Q}) := \sup_{B\in\mathcal{B}} |\mathbb{P}(B) - \mathbb{Q}(B)| \tag{6}$$

of probability measures $\mathbb{P}$ and $\mathbb{Q}$ on Ξ for a given system $\mathcal{B}$ of Borel subsets of Ξ. This is due to the estimate

$$\zeta_{r,\mathcal{B}}(\mathbb{P},\mathbb{Q}) \le C\alpha_{\mathcal{B}}(\mathbb{P},\mathbb{Q})^{\frac{1}{s+1}} \tag{7}$$

for all probability measures $\mathbb{P}$ and $\mathbb{Q}$ on Ξ with supports contained in the ball $\mathbb{B}(0,R)$ which may be derived as [25, Corollary 3.2]. The constant $C = C(R)$ only depends on the problem (1) and the radius R. Since we are interested in measuring the distance of two discrete probability measures $\mathbb{P}$ and $\mathbb{Q}$ with $\mathbb{Q}$'s support being a subset of the support of $\mathbb{P}$, both supports are contained in some ball around zero and, thus, the estimate (7) applies. As shown in [25, Proposition 3.1] (see also [23, Proposition 1]), the class $\mathcal{B}$ of polyhedra has to contain all sets of the form

$$\{\xi \in \Xi : h(\xi) \in Tx + B\} = \{\xi \in \Xi : \xi \in h^{-1}(Tx + B)\}\,, \tag{8}$$

where h is an affine mapping from $\mathbb{R}^s$ to $\mathbb{R}^d$, $x \in X$ and B is a polyhedron each of whose facets, i.e., $(d-1)$-dimensional faces, is parallel to a facet of the cone $\operatorname{pos} W := \{Wy_1 : y_1 \in \mathbb{R}^{m_1}_+\}$ or of the unit cube $[0,1]^d$.

In this paper, we consider the particular instance of the sets (8) with $d = s$ and $h(\xi) = \xi$ which corresponds to the situation of mixed-integer two-stage stochastic programs with random right-hand side. The corresponding class $\mathcal{B}$ of polyhedra in Ξ is denoted by $\mathcal{B}_{\text{poly(W)}}$ and the polyhedral discrepancy by $\alpha_{\mathcal{B}_{\text{poly(W)}}}$. If every facet of $\operatorname{pos} W$ parallels a facet of the unit cube, $\alpha_{\mathcal{B}_{\text{poly(W)}}}$ coincides with the rectangular discrepancy $\alpha_{\mathcal{B}_{\text{rect}}}$ that is defined by (6) and the set $\mathcal{B}_{\text{rect}}$ of all rectangles in $\mathbb{R}^s$. The latter discrepancy becomes suitable in case of pure integer recourse, i.e.,

$$\Phi(u,t) := \min\{\langle u, y\rangle : \bar{W}y \le t, y \in \mathbb{Z}^{m_2}_+\}\,.$$

Thus, when studying the polyhedral discrepancy for arbitrary matrices W in the following, the rectangular case has not to be considered separately.

The following simple example illustrates the stability results of [23] and [25] and highlights the relevance of $\alpha_{\mathcal{B}_{\text{poly(W)}}}$.

Example 1.1. *Let $\Xi := [0,1]^2$, $X := \{0\} \in \mathbb{R}^2$ and $\mathbb{P}$ be some probability measure on Ξ. We consider the optimal value of the following mixed-integer two-stage stochastic program:*

$$v(\mathbb{P}) := \inf_{x \in X} \int_{[0,1]^2} \Phi(\xi - x)\,\mathbb{P}(d\xi) = \int_{[0,1]^2} \Phi(\xi)\,\mathbb{P}(d\xi),$$

$$\Phi(\xi) := \min\{y_1(\xi) + 2y_2(\xi) : y_1(\xi) + y_2(\xi) \ge \xi_1, y_2(\xi) \le \xi_2, y_1 \in \mathbb{Z}_+, y_2 \in \mathbb{R}_+\}. \tag{9}$$

$\Phi(\xi)$ is equal to 1 if $\xi_1 > \min\{0.5, \xi_2\}$ and equal to $2\xi_1$ otherwise. A plot of $\xi \mapsto \Phi(\xi)$ is presented in Figure 1.1.

Assuming that $\mathbb{P}$ follows a uniform distribution on the line segment $\{(z,z) : z \in (0.1, 0.4)\}$, we define for every $\varepsilon > 0$ the shifted measure $\mathbb{P}_\varepsilon$ via $\mathbb{P}_\varepsilon(A) := \mathbb{P}(A + \binom{\varepsilon}{-\varepsilon})$ for every Borel set $A \subset \Xi$. It follows that $v(\mathbb{P}_\varepsilon) - v(\mathbb{P}) = 0.5$ for every $\varepsilon > 0$. On the other hand, with $\varepsilon \searrow 0$ the measures $\mathbb{P}_\varepsilon$ converge to $\mathbb{P}$ weakly as well as with respect to the rectangular discrepancy $\alpha_{\mathcal{B}_{rect}}$.

Writing problem (9) in standard form (2), the non-integer part of y is assigned to the recourse matrix

$$W = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

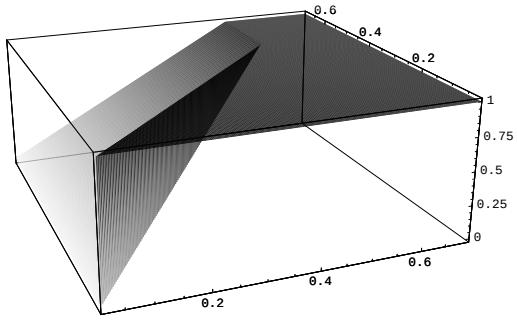

Figure 1: Cost function $\xi \mapsto \Phi(\xi)$ from Example 1.1.

With regard to the aforementioned continuity of $v(\cdot)$ w.r.t. $\alpha_{\mathcal{B}_{poly(W)}}$, it is not surprising that $\alpha_{\mathcal{B}_{poly(W)}}(\mathbb{P}, \mathbb{P}_\varepsilon) = 1$ for every $\varepsilon > 0$.
Note that it has been shown in [16], that, under certain conditions, $\alpha_{\mathcal{B}_{poly(W)}}$ can be estimated against $\alpha_{\mathcal{B}_{rect}}$. However, in our case $\mathbb{P}$ is not absolutely continuous w.r.t. the Lebesgue measure on $\Xi \subset \mathbb{R}^2$, and, hence, the result in [16] does not apply.

It is worth mentioning that polyhedral discrepancies also become suitable for chance constrained stochastic programs of the form

$$\min\{\langle c, x\rangle : x \in X,\, \mathbb{P}(Tx \geq h(\xi)) \geq p\}, \tag{10}$$

where $p \in [0, 1]$ and c, X, T and $h(\cdot)$ are defined as above. The chance constraint may be rewritten as

$$\mathbb{P}(Tx \geq h(\xi)) = \mathbb{P}(\{\xi \in \mathbb{R}^s : h(\xi) \in (-\infty, Tx]\}) = \mathbb{P}(h^{-1}(-\infty, Tx]),$$

and, hence, under certain regularity conditions (see, e.g., [22, Section 3.3]), optimal values and solution sets of (10) behave continuous with respect to the discrepancy $\alpha_{\mathcal{B}}$ if $\mathcal{B}$ contains all polyhedra of the form $h^{-1}(-\infty, Tx]$ for $x \in X$.

The present paper is organized as follows. In Section 2 we state the problem of optimal scenario reduction with respect to a discrepancy distance $\alpha_{\mathcal{B}}$ and decompose it into a combinatorial and a linear optimization problem. Extending our earlier work in [9], we discuss in Section 3 how the coefficients of the linear program may be computed in case of the polyhedral discrepancy $\alpha_{\mathcal{B}_{\mathrm{poly(W)}}}$. Algorithms for determining the optimally reduced probability distribution (with respect to $\alpha_{\mathcal{B}_{\mathrm{poly(W)}}}$) are developed in Sections 4 and 5. In Section 6 we provide and discuss numerical results.

2 Scenario Reduction

We consider a probability measure $\mathbb{P}$ with finite support $\{\xi^1, \ldots, \xi^N\}$ and set $p_i := \mathbb{P}(\{\xi^i\}) > 0$ for $i = 1, \ldots, N$. Denoting by δ_ξ the Dirac-measure placing mass one

at the point ξ, the measure $\mathbb{P}$ has the form

$$\mathbb{P} = \sum_{i=1}^{N} p_i \delta_{\xi^i} \,. \tag{11}$$

The problem of optimal scenario reduction consists of determining a discrete probability measure $\mathbb{Q}$ on $\mathbb{R}^s$ supported by a subset of $\{\xi^1, \ldots, \xi^N\}$ and deviating from $\mathbb{P}$ as little as possible with respect to some distance, in this case a certain discrepancy $\alpha_{\mathcal{B}}$. It can be written as

$$\begin{aligned} &\text{minimize } \alpha_{\mathcal{B}}(\mathbb{P}, \mathbb{Q}) = \alpha_{\mathcal{B}}\Big(\sum_{i=1}^{N} p_i \delta_{\xi^i}, \sum_{j=1}^{n} q_j \delta_{\eta^j} \Big) \\ &\text{subject to } \{\eta^1, \ldots, \eta^n\} \subset \{\xi^1, \ldots, \xi^N\}, q_j \geq 0 \, (j = 1, \ldots, n), \sum_{j=1}^{n} q_j = 1. \end{aligned} \tag{12}$$

The variables to be optimally adjusted here are the support $\eta = \{\eta^1, \ldots, \eta^n\}$ and the probability weights $q = (q_1, \ldots, q_n)$ of the reduced measure $\mathbb{Q}$, altogether they define $\mathbb{Q}$ via

$$\mathbb{Q} = \sum_{j=1}^{n} q_j \delta_{\eta^j}. \tag{13}$$

The optimization problem (12) may be decomposed into an outer problem for determining $\operatorname{supp} \mathbb{Q} = \eta$ and an inner problem for choosing the probabilities q_j. To this end, we denote by $\alpha_{\mathcal{B}}(\mathbb{P}, (\eta, q))$ the $\mathcal{B}$-discrepancy between $\mathbb{P}$ and $\mathbb{Q}$, and by S_n the standard simplex in $\mathbb{R}^n$:

$$\begin{aligned} \alpha_{\mathcal{B}}(\mathbb{P}, (\eta, q)) \;&:=\; \alpha_{\mathcal{B}}\Big(\sum_{i=1}^{N} p_i \delta_{\xi^i}, \sum_{j=1}^{n} q_j \delta_{\eta^j} \Big) \\ S_n \;&:=\; \Big\{ q \in \mathbb{R}^n \,:\, q_j \geq 0,\, j = 1, \ldots, n,\, \sum_{j=1}^{n} q_j = 1 \Big\}. \end{aligned}$$

Using this notation, the scenario reduction problem (12) can be written as

$$\min_{\eta} \{ \inf_{q \in S_n} \alpha_{\mathcal{B}}(\mathbb{P}, (\eta, q)) \,:\, \eta \subset \{\xi^1, \ldots, \xi^N\}, \#\eta = n \}, \tag{14}$$

with the inner problem

$$\min\{ \alpha_{\mathcal{B}}(\mathbb{P}, (\eta, q)) \,:\, q \in S_n \} \tag{15}$$

for the fixed support η. The combinatorial optimization problem (14) is addressed in Section 5. In the remaining part of this section, we introduce some notation and recall [9, Section 3.1] to show how the inner problem (15) can be formulated as a linear optimization problem and how the dimensionality of the latter can be reduced.

When adressing the inner problem (15) for given η we may assume for the sake of notational simplicity, that $\eta = \{\xi^1, \ldots, \xi^n\}$. Then, (15) is of the form:

$$\begin{aligned} &\text{minimize } \alpha_{\mathcal{B}}(\mathbb{P}, (\{\xi^1, \ldots, \xi^n\}, q)) = \alpha_{\mathcal{B}}\Big(\sum_{i=1}^{N} p_i \delta_{\xi^i}, \sum_{j=1}^{n} q_j \delta_{\xi^j} \Big) \\ &\text{subject to } q \in S_n. \end{aligned} \tag{16}$$

The finiteness of $\mathbb{P}$'s support allows to define for $B \in \mathcal{B}$ the *critical index set* $I(B)$ through the relation

$$I(B) := \{i \in \{1, \dots, N\} : \ \xi^i \in B\}$$

and to write

$$|\mathbb{P}(B) - \mathbb{Q}(B)| = \left| \sum_{i \in I(B)} p_i - \sum_{j \in I(B) \cap \{1,\dots,n\}} q_j \right|. \tag{17}$$

Furthermore, we define the *system of critical index sets* of a system of Borel sets $\mathcal{B}$ as

$$\mathcal{I}_{\mathcal{B}} := \{I(B) \ : \ B \in \mathcal{B}\}.$$

Thus, the $\mathcal{B}$-discrepancy between $\mathbb{P}$ and $\mathbb{Q}$ can be reformulated as follows:

$$\alpha_{\mathcal{B}}(\mathbb{P}, \mathbb{Q}) = \max_{I \in \mathcal{I}_{\mathcal{B}}} \left| \sum_{i \in I} p_i - \sum_{j \in I \cap \{1,\dots,n\}} q_j \right|. \tag{18}$$

This allows to solve (16) by means of the following linear optimization problem:

$$\begin{array}{lllll} \text{minimize} & t & \text{subject to} & q \in S_n, & \\ & \left. \begin{array}{lll} -\sum_{j \in I \cap \{1,\dots,n\}} q_j & \leq & t - \sum_{i \in I} p_i \\ \sum_{j \in I \cap \{1,\dots,n\}} q_j & \leq & t + \sum_{i \in I} p_i \end{array} \right\} I \in \mathcal{I}_{\mathcal{B}}. \end{array} \tag{19}$$

The number of inequalities may be too large to solve (19) numerically, but whenever two critical index sets share the same intersection with the set $\{1, \dots, n\}$, only the right-hand sides of the related inequalities differ. Thus, it is possible to pass to the minimum of all right-hand sides corresponding to the same left-hand side. To this end, we introduce the following *reduced system of critical index sets*

$$\mathcal{I}^*_{\mathcal{B}} := \{I(B) \cap \{1, \dots, n\} \ : \ B \in \mathcal{B}\}.$$

Thereby, every member $J \in \mathcal{I}^*_{\mathcal{B}}$ of the reduced system is associated with a familiy $\varphi(J) \subset \mathcal{I}_{\mathcal{B}}$ of critical index sets, all of which share the same intersection with $\{1, \dots, n\}$:

$$\varphi(J) := \{I \in \mathcal{I}_{\mathcal{B}} \ : \ J = I \cap \{1, \dots, n\}\} \quad (J \in \mathcal{I}^*_{\mathcal{B}}). \tag{20}$$

Finally, we consider the quantities

$$\gamma^J := \max_{I \in \varphi(J)} \sum_{i \in I} p_i \quad \text{and} \quad \gamma_J := \min_{I \in \varphi(J)} \sum_{i \in I} p_i \quad (J \in \mathcal{I}^*_{\mathcal{B}}), \tag{21}$$

to write problem (19) as

$$\begin{array}{lllll} \text{minimize} & t & \text{subject to} & q \in S_n, & \\ & \left. \begin{array}{lll} -\sum_{j \in J} q_j & \leq & t - \gamma^J \\ \sum_{j \in J} q_j & \leq & t + \gamma_J \end{array} \right\} J \in \mathcal{I}^*_{\mathcal{B}}. \end{array} \tag{22}$$

Since $|\mathcal{I}_{\mathcal{B}}| \leq 2^N$ and $|\mathcal{I}^*_{\mathcal{B}}| \leq 2^n$, passing from (19) to (22) indeed drastically reduces the maximum number of inequalities and can make problem (15) tractable for numerical solutions.

3 Determining the coefficients

In our earlier work [9, Section 3.2], it is described how the coefficients $\mathcal{I}^*_{\mathcal{B}}, \gamma^J, \gamma_J$ can be determined computationally in case of the cell discrepancy (where $\mathcal{B} = \{\xi + \mathbb{R}^s_- : \xi \in \mathbb{R}^s\}$). In this section, this approach is extended to the more general polyhedral discrepancies.

Given the recourse matrix W, we consider k pairwise linearly independent vectors $m^1, \ldots, m^k$ in $\mathbb{R}^s$ such that every facet of $\operatorname{pos} W$ and of the unit cube $[0,1]^s$ is normal relative to m^i for one $i \in \{1, \ldots, k\}$. By M we denote the $(k \times s)-$matrix whose rows are given by $m^1, \ldots, m^k$, respectively. Then, due to its special form, every polyhedron B in $\mathcal{B}_{\text{poly(W)}}$ can be written as

$$B = \left\{x \in \mathbb{R}^s : \underline{a}^B \leq Mx \leq \bar{a}^B\right\} \tag{23}$$

for suitable $k-$dimensional vectors $\underline{a}^B$ and $\bar{a}^B$ with $\underline{a}^B_i \leq \bar{a}^B_i \in \mathbb{R} \cup \{+\infty, -\infty\}$ for $i = 1, \ldots, k$. Since B is determined by the vectors $\underline{a}^B$ and $\bar{a}^B$, we will use the notation $\left[\underline{a}^B, \bar{a}^B\right]$ for B in the following.
Analogous to the concept of *supporting cells* in [9], we will show that it suffices to consider the following *supporting polyhedra*. Loosely speaking, a polyhedron $B \in \mathcal{B}_{\text{poly(W)}}$ is called supporting if each of its facets contains an element of $\{\xi^1, \ldots, \xi^n\}$ in a way that B can not be enlarged without changing the intersection of B's interior with $\{\xi^1, \ldots, \xi^n\}$, cf. Figure 2. This is formalized by the following definitions. We introduce the sets

$$\begin{aligned} R_j &:= \left\{\left\langle m^j, \xi^i \right\rangle, i = 1, \ldots, n\right\} \cup \{\infty, -\infty\}\ ,\ j = 1, \ldots, k, \text{ and} \\ R &:= \times_{j=1}^k R_j. \end{aligned} \tag{24}$$

Then, every polyhedron $B = [\underline{a}^B, \bar{a}^B] \in \mathcal{B}_{\text{poly(W)}}$ with

$$\underline{a}^B, \bar{a}^B \in R \tag{25}$$

admits a further representation by two vectors $\underline{i} := (\underline{i}_1, \ldots, \underline{i}_k)$, $\bar{i} := (\bar{i}_1, \ldots, \bar{i}_k)$ with $\underline{i}_j, \bar{i}_j \in \{1, \ldots, n, \pm\infty\}$

$$\underline{a}^B_j = \left\langle m^j, \xi^{\underline{i}_j}\right\rangle \quad \text{and} \quad \bar{a}^B_j = \langle m^j, \xi^{\bar{i}_j}\rangle,$$

where we set for notational convenience $\langle m^j, \xi^{\pm\infty}\rangle := \pm\infty$, respectively. Condition (25) ensures that every hyperplane in $\mathbb{R}^s$ including a facet of B contains an element of $\{\xi^1, \ldots, \xi^n\}$.

Definition 3.1. *A polyhedron $B \in \mathcal{B}_{poly(W)}$ with* (25) *is called* supporting*, if it admits a representation $\underline{i}, \bar{i}$, such that for every $j, l \in \{1, \ldots, k\}, j \neq l$*

$$\begin{aligned} \langle m^j, \xi^{\underline{i}_j}\rangle < \langle m^j, \xi^{\underline{i}_l}\rangle &< \langle m^j, \xi^{\bar{i}_j}\rangle \textit{ whenever } \underline{i}_l \neq \pm\infty, \textit{ and} \\ \langle m^j, \xi^{\underline{i}_j}\rangle < \langle m^j, \xi^{\bar{i}_l}\rangle &< \langle m^j, \xi^{\bar{i}_j}\rangle \textit{ whenever } \bar{i}_l \neq \pm\infty. \end{aligned} \tag{26}$$

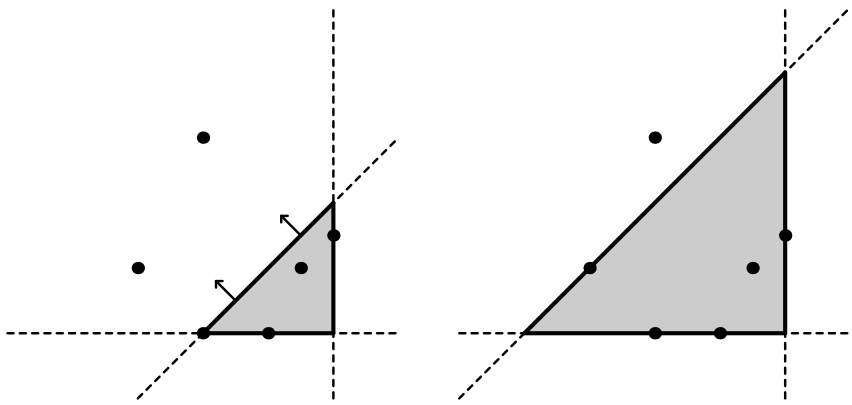

Figure 2: Non supporting polyhedron (left) and supporting polyhedron (right). The dots represent the remaining scenarios $\xi^1, \ldots, \xi^n$.

The set of all supporting polyhedra is defined by

$$\mathcal{P} := \{B \subset \mathbb{R}^s : B \textit{ is a supporting polyhedron}\}.$$

The following proposition parallels [9, Prop. 3.1] and shows that every critical index set J corresponds to a maximal supporting polyhedron B whose interior does not contain any ξ^i with $i \in \{1, \ldots, n\} \setminus J$.

Proposition 3.2. *For any $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ there exists a supporting polyhedron B such that $\gamma^J = \mathbb{P}(\text{int } B)$ and*

$$\cup_{j \in J}\{\xi^j\} = \{\xi^1, \ldots, \xi^n\} \cap \text{int } B. \tag{27}$$

Proof. We consider an arbitrary $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$. From the definition of $\varphi(J)$ in (20) it follows that for any $I \in \varphi(J)$ there exists some $C \in \mathcal{B}_{poly(W)}$ such that $I = I(C)$ and $J = I(C) \cap \{1, \ldots, n\}$. By definition of $I(C)$ we have

$$\sum_{i \in I} p_i = \sum_{i \in I(C)} p_i = \mathbb{P}(C),$$

and, hence,

$$\begin{aligned}
\gamma^J &= \max_{I \in \varphi(J)} \sum_{i \in I} p_i = \max\left\{\mathbb{P}(C) \,:\, C \in \mathcal{B}_{poly(W)},\ J = I(C) \cap \{1, \ldots, n\}\right\} \\
&= \max\left\{\mathbb{P}(C) \,:\, C \in \mathcal{B}_{poly(W)},\ C \cap \{\xi^1, \ldots, \xi^n\} = \cup_{j \in J}\{\xi^j\}\right\} \\
&= \max\left\{\mathbb{P}([\underline{a}, \bar{a}]) \,:\, \underline{a}, \bar{a} \in (\mathbb{R} \cup \{\pm\infty\})^k\,,\right. \\
&\qquad \left.[\underline{a}, \bar{a}] \cap \{\xi^1, \ldots, \xi^n\} = \cup_{j \in J}\{\xi^j\}\right\}.
\end{aligned} \tag{28}$$

Let $\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right]$ be a polyhedron attaining the maximum, i.e., $\gamma^J = \mathbb{P}(\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right])$ and $\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right] \cap \{\xi^1, \ldots, \xi^n\} = \cup_{j\in J}\{\xi^j\}$. In addition, due to finiteness of the set $\left\{\xi^1, \ldots, \xi^N\right\}$, we can assume that these identities are also valid for $\operatorname{int}\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right]$, i.e.

$$\gamma^J = \mathbb{P}(\operatorname{int}\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right]) \quad \text{and} \quad \operatorname{int}\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right] \cap \{\xi^1, \ldots, \xi^n\} = \cup_{j\in J}\{\xi^j\}. \tag{29}$$

In the following, we will enlarge $\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right]$ by succesively shifting its facets until it becomes supporting. To this end, we put

$$\underline{a}(t) := (\underline{a}_1^{(0)} - t, \underline{a}_2^{(0)}, \ldots, \underline{a}_k^{(0)})\,,\ t \geq 0,$$

and consider the polyhedral enlargement $\left[\underline{a}(t), \bar{a}^{(0)}\right]$ of the polyhedron $\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right]$. We set

$$\tau := \sup\left\{t \geq 0 \,:\, \operatorname{int}\left[\underline{a}(t), \bar{a}^{(0)}\right] \cap \left\{\xi^1, \ldots, \xi^n\right\} = \operatorname{int}\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right] \cap \left\{\xi^1, \ldots, \xi^n\right\}\right\}.$$

In particular, $\tau = \infty$ holds if $\underline{a}_1^{(0)} = -\infty$ and whenever $\tau = \infty$ we define $\underline{i}_1 := -\infty$. If $\tau < \infty$ there exists $\underline{i}_1 \in \{1, \ldots, n\} \setminus J$ such that $\langle m^1, \xi^{\underline{i}_1}\rangle = \underline{a}_1^{(0)} - \tau$ and

$$\underline{a}_j^{(0)} < \langle m^j, \xi^{\underline{i}_1}\rangle < \bar{a}_j^{(0)} \text{ for } j \neq 1.$$

Indeed, this is true since one can find a $\xi^{\underline{i}_1}$ with $\underline{i}_1 \in \{1, \ldots, n\} \setminus J$ that lies in the interior of $\left[\underline{a}(\tau + \varepsilon), \bar{a}^{(0)}\right]$ for $\varepsilon > 0$.
We put $\underline{a}^{(1)} := \underline{a}(\tau)$ and consider now, in the second step, the enlarged polyhedron $\left[\underline{a}^{(1)}, \bar{a}^{(0)}\right]$, still fulfilling the second identity of (29). Hence, since $\left[\underline{a}^{(0)}, \bar{a}^{(0)}\right]$ was maximizing, $\left[\underline{a}^{(1)}, \bar{a}^{(0)}\right]$ fulfils the first identity of (29), too.
We repeat the above construction for the coordinate $\bar{a}^{(0)}$ by defining

$$\begin{aligned}
\bar{a}(t) &:= (\bar{a}_1^{(0)} + t, \bar{a}_2^{(0)}, \ldots, \bar{a}_k^{(0)}), \quad (t \geq 0)\\
\tau &:= \sup\left\{t \geq 0 : \operatorname{int}\left[\underline{a}^{(1)}, \bar{a}(t)\right] \cap \left\{\xi^1, \ldots, \xi^n\right\} = \operatorname{int}\left[\underline{a}^{(1)}, \bar{a}^{(0)}\right] \cap \left\{\xi^1, \ldots, \xi^n\right\}\right\}\\
\bar{a}^{(1)} &:= \bar{a}(\tau).
\end{aligned}$$

Again, if $\tau < \infty$, there exists $\bar{i}_1 \in \{1, \ldots, n\}$ such that $\langle m^1, \xi^{\bar{i}_1}\rangle = \bar{a}_1^{(0)} + \tau$ and $\underline{a}_j^{(0)} < \langle m^j, \xi^{\bar{i}_1}\rangle < \bar{a}_j^{(0)}$ for $j \neq 1$. Otherwise, we put $\bar{i}_1 = \infty$.
Continuing the construction in this way for the coordinates $2, \ldots, k$, we arrive at the polyhedron $B := \left[\underline{a}^{(k)}, \bar{a}^{(k)}\right]$ with (25) and (27) as well as the indices $\underline{i}_l, \bar{i}_l$ for $l = 1, \ldots, k$ with $\underline{a}_j^{(k)} = \underline{a}_j^{(j)} = \langle m^j, \xi^{\underline{i}_j}\rangle, \bar{a}_j^{(k)} = \bar{a}_j^{(j)} = \langle m^j, \xi^{\bar{i}_j}\rangle$. Furthermore, one obtains the estimates

$$\begin{aligned}
\underline{a}_j^{(l-1)} < \langle m^j, \xi^{\underline{i}_l}\rangle &< \underline{a}_j^{(l-1)} \text{ for } j \neq l \text{ whenever } \underline{i}_l \neq \pm\infty, \text{ and}\\
\underline{a}_j^{(l-1)} < \langle m^j, \xi^{\bar{i}_l}\rangle &< \underline{a}_j^{(l-1)} \text{ for } j \neq l \text{ whenever } \bar{i}_l \neq \pm\infty.
\end{aligned}$$

These inequalities remain valid for the final vectors $\underline{a}^{(k)}, \bar{a}^{(k)}$, due to $\underline{a}_j^{(l-1)} \geq \underline{a}_j^{(k)}$ and $\bar{a}_j^{(l-1)} \leq \bar{a}_j^{(k)}$ for $j = 1, \ldots, k$. Thus, (26) holds and B is supporting. Since both identities of (29) remain valid during the construction of B, it follows that B possesses the asserted properties. □

Corollary 3.3. *The following identities hold:*

$$\begin{aligned}\mathcal{I}^*_{\mathcal{B}_{poly(W)}} &= \{J \subseteq \{1,\dots,n\} \,:\, \exists B \in \mathcal{P} \text{ such that (27) holds true}\},\\ \gamma^J &= \max\{\mathbb{P}(\text{int } B) \,:\, B \in \mathcal{P},\ (27) \text{ holds true}\} \quad \forall J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}.\end{aligned}$$

Proof. The inclusion '$\subseteq$' in the first identity and the inequality '$\leq$' in the second identity follow directly from Proposition 3.2. For the reverse direction of the first identity, let $B = \left[\underline{a}^B, \bar{a}^B\right] \in \mathcal{P}$ be given such that (27) holds true for some $J \subseteq \{1,\dots,n\}$. Due to finiteness of $\{\xi^1,\dots,\xi^n\}$ there exists an $\varepsilon > 0$ such that

$$\{\xi^1,\dots,\xi^n\} \cap \left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right] = \{\xi^1,\dots,\xi^n\} \cap \text{int } \left[\underline{a}^B, \bar{a}^B\right] = \cup_{j\in J}\{\xi^j\}, \tag{30}$$

where each entry of the vector $\bar{\varepsilon} \in \mathbb{R}^k$ is equal to ε. Since $\left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right] \in \mathcal{B}_{poly(W)}$, we observe

$$\begin{aligned}I\left(\left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right]\right) &= \{i \in \{1,\dots,N\} \,:\, \xi^i \in \left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right]\}\\ &= J \cup \{i \in \{n+1,\dots,N\} \,:\, \xi^i \in \left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right]\}.\end{aligned}$$

Therefore,

$$I\left(\left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right]\right) \cap \{1,\dots,n\} = J \cap \{1,\dots,n\} = J, \tag{31}$$

which provides $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ via the definition of $\mathcal{I}^*_{\mathcal{B}_{poly(W)}}$. Consequently, also inclusion '$\supseteq$' in the first identity holds true. To verify the relation '$\geq$' in the second identity, let $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ and $B = \left[\underline{a}^B, \bar{a}^B\right] \in \mathcal{P}$ be arbitrary, such that (27) holds true. We choose an $\varepsilon > 0$ with

$$\{\xi^1,\dots,\xi^N\} \cap \left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right] = \{\xi^1,\dots,\xi^N\} \cap \text{int } \left[\underline{a}^B, \bar{a}^B\right] \tag{32}$$

and conclude (31). Therefore, $I\left(\left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right]\right) \in \varphi(J)$ (see (20)) and

$$\begin{aligned}\gamma^J &\geq \sum_{i \in I([\underline{a}^B+\bar{\varepsilon}, \bar{a}^B-\bar{\varepsilon}])} p_i = \sum_{\xi^i \in [\underline{a}^B+\bar{\varepsilon}, \bar{a}^B-\bar{\varepsilon}]} \mathbb{P}(\{\xi^i\}) = \mathbb{P}\left(\left[\underline{a}^B + \bar{\varepsilon}, \bar{a}^B - \bar{\varepsilon}\right]\right)\\ &= \mathbb{P}(\text{int } \left[\underline{a}^B, \bar{a}^B\right]),\end{aligned}$$

where the last identity follows from (32). This proves the inequality '$\geq$' in the second identity, since $\left[\underline{a}^B, \bar{a}^B\right] \in \mathcal{P}$ was chosen arbitrarily such that (27) holds true. □

From Corollary 3.3 it follows that the set $\mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ and the upper coefficients γ^J for $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ can be determined, whenever one knows the system of $\mathcal{P}$ of supporting polyhedra. The following proposition shows how the lower coefficients γ_J for $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ may be computed.

Proposition 3.4. *For all $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$, one has $\gamma_J = \sum_{i\in I} p_i$, where I is given by*

$$I := \{i \in \{1,\dots,N\} \,:\, \min_{j\in J}\langle m^l, \xi^j\rangle \le \langle m^l, \xi^i\rangle \le \max_{j\in J}\langle m^l, \xi^j\rangle \,, l = 1,\dots,k\}.$$

Proof. We consider an arbitrary $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$. Completely analogous to (28) in the proof of Proposition 3.2, it follows that

$$\gamma_J = \min\{\mathbb{P}([\underline{a}, \bar{a}]) \,:\, [\underline{a}, \bar{a}] \cap \{\xi^1,\dots,\xi^n\} = \cup_{j\in J}\{\xi^j\}\}. \tag{33}$$

We define $\underline{a}^*, \bar{a}^* \in \mathbb{R}^k$ by

$$\begin{aligned} \underline{a}^*_l &:= \min_{j\in J}\langle m^l, \xi^j\rangle \text{ for } l = 1,\dots,k, \text{ and} \\ \bar{a}^*_l &:= \max_{j\in J}\langle m^l, \xi^j\rangle \text{ for } l = 1,\dots,k, \end{aligned}$$

to obtain $\xi^j \in [\underline{a}^*, \bar{a}^*]$ for all $j \in J$ and, therefore, $\cup_{j\in J}\{\xi^j\} \subseteq [\underline{a}^*, \bar{a}^*] \cap \{\xi^1,\dots,\xi^n\}$. If this inclusion is strict, there is some $i \in \{1,\dots,n\}\backslash J$ such that $\xi^i \in [\underline{a}^*, \bar{a}^*]$. From $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ it follows the existence of some $B \in \mathcal{B}_{poly(W)}$ with $J = I(B) \cap \{1,\dots,n\}$. Thus, we obtain $\xi^j \in B$ for all $j \in J$, which entails that $[\underline{a}^*, \bar{a}^*] \subseteq B$, by construction of $[\underline{a}^*, \bar{a}^*]$. We derive that $\xi^i \in B$ and, hence, $i \in I(B)$. On the other hand, $i \in \{1,\dots,n\}\backslash J$, which is a contradiction. It follows that

$$\cup_{j\in J}\{\xi^j\} = [\underline{a}^*, \bar{a}^*] \cap \{\xi^1,\dots,\xi^n\}$$

and, thus, $\gamma_J \le \mathbb{P}([\underline{a}^*, \bar{a}^*])$. On the other hand, consider arbitrary $\underline{a}$ and $\bar{a}$ with $[\underline{a}, \bar{a}] \cap \{\xi^1,\dots,\xi^n\} = \cup_{j\in J}\{\xi^j\}$. Then, $\xi^j \in [\underline{a}, \bar{a}]$ for all $j \in J$, and by construction of $[\underline{a}^*, \bar{a}^*]$ it follows that $[\underline{a}^*, \bar{a}^*] \subseteq [\underline{a}, \bar{a}]$. Consequently, $\mathbb{P}([\underline{a}^*, \bar{a}^*]) \le \mathbb{P}([\underline{a}, \bar{a}])$ holds. Passing to the minimum over all such $\underline{a}, \bar{a}$ and applying identity (33) provides $\mathbb{P}([\underline{a}^*, \bar{a}^*]) \le \gamma_J$. Finally, the assertion follows from

$$\gamma_J = P([\underline{a}^*, \bar{a}^*]) = \sum_{\xi^i \in [\underline{a}^*, \bar{a}^*]} p_i = \sum_{i\in I} p_i.$$

□

With Corollary 3.3 and Proposition 3.4 at hand, we propose in the next section an algorithm to calculate the coefficients of (22) and to solve the inner problem (15).

4 Optimal redistribution algorithm

For using numerically the concept of supporting polyhedra, one has to determine the set R defined by (24). Thus, given the matrix W, one has to identify a normal

vector for each facet of the convex cone $\operatorname{pos} W$. This transformation of a *vertices-based* representation of $\operatorname{pos} W$ to one based on halfspaces is a well-studied problem for which efficient algorithms are available, e.g. the implementation *lrs* ([1]) of [2]'s reverse search algorithm, and the implemenation *cdd+* ([4]) of [13]'s double description method.
Corollary 3.3 and Proposition 3.4 show that the coefficients of the linear inner problem (15), i.e. $\mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ and γ^J, γ_J for $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$, can be determined by iterating through the set $\mathcal{P}$ of supporting polyhedra. With regard to the huge number of potential supporting polyhedra, we propose for this iteration a recursive approach. More precisely, the supporting polyhedra $[\underline{a}, \bar{a}] = [(\underline{a}_j)_{j=1}^k, (\bar{a}_j)_{j=1}^k]$ are constructed recursively for $j = 1, \ldots, k$, while ensuring at every step j that condition (26) is still fulfilled when the $j-th$ coordinate is added. This is done within FUNCTION **Iterate** of the following

Algorithm 4.1. *Optimal redistribution.*

Step [0]: *Put* $\mathcal{I}^*_{\mathcal{B}_{poly(W)}} = \{\varnothing\}$, *and* $\gamma^J = 0$ *for all* $J \subseteq \{1, \ldots, n\}$.
Set $J = \{1, \ldots, n\}$ *and* $\bar{I} = \{1, \ldots, N\}$.

Step [1]: *Call* **Iterate**$(0, 0, 0, J, \bar{I})$.

Step [2]: *With the additional data* $\mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ *and* γ_J, γ^J *for all* $J \in \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ *solve the linear optimization problem* (22).

FUNCTION **Iterate**$(l, (\underline{i}_j)_{j=1}^l, (\bar{i}_j)_{j=1}^l, J, \bar{I})$:

IF $l = k$ THEN
call **UpdateData**$((\underline{i}_j)_{j=1}^l, (\bar{i}_j)_{j=1}^l, J, \bar{I})$ and RETURN.
Set $l = l + 1$.
FOR $\underline{i}_l = 1, \ldots, n+1$ and $\bar{i}_l = 1, \ldots, n+1$:
IF (26) does not hold for every $j \in \{1, \ldots, l-1\}, i = l$,
and for every $i \in \{1, \ldots, l-1\}, j = l$, THEN CONTINUE<FOR>.
Set

$$\underline{a}_l = \begin{cases} \langle m^l, \xi^{\underline{i}_l} \rangle & \text{if } \underline{i}_l \in \{1, \ldots, n\}, \\ -\infty & \text{if } \underline{i}_l = n+1, \end{cases}$$

$$\bar{a}_l = \begin{cases} \langle m^l, \xi^{\bar{i}_l} \rangle & \text{if } \bar{i}_l \in \{1, \ldots, n\}, \\ +\infty & \text{if } \bar{i}_l = n+1. \end{cases}$$

Update $\bar{I} = \bar{I} \cap \{i \in \{1, \ldots, N\} : \underline{a}_l < \langle m^l, \xi^i \rangle < \bar{a}_l)\}$.
IF $\bar{I} = \varnothing$ THEN CONTINUE<FOR>.
Update $J = J \cap \bar{I}$.
Call **Iterate**$(l, (\underline{i}_j)_{j=1}^l, (\bar{i}_j)_{j=1}^l, J, \bar{I})$.
END(FOR).
RETURN.

FUNCTION **UpdateData**$((\underline{i}_j)_{j=1}^k, (\bar{i}_j)_{j=1}^k, J, \bar{I})$:

IF $J \notin \mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ THEN

Update $\mathcal{I}^*_{\mathcal{B}_{poly(W)}} = \mathcal{I}^*_{\mathcal{B}_{poly(W)}} \cup \{J\}$.

Set $\gamma_J = \sum_{i \in I} p_i$ with

$$I = \Big\{ i \in \bar{I} : \min_{j \in J} \langle m^l, \xi^j \rangle \leq \langle m^l, \xi^i \rangle \leq \max_{j \in J} \langle m^l, \xi^j \rangle \, , l = 1, \ldots, k \Big\}.$$

END(IF).

Update $\gamma^J = \max \Big\{ \gamma^J, \sum_{i \in \bar{I}} p_i \Big\}$.

RETURN.

Remark 4.2. *When using Algorithm 4.1 repeatedly with varying support, e.g. within one of the algorithms mentioned in the next section, it would be desirable to decrease the numerical complexity by using some of the data* $\mathcal{I}^*_{\mathcal{B}_{poly(W)}}$ *and* γ_J, γ^J *computed for another support. While this is possible for* γ^J *and* γ_J *within Algorithm 4.1. of [9], it is unfeasible inside our Algorithm 4.1, since* γ^J *and* γ_J *are already determined by the set* $\bar{I}$*, that is calculated simultaneously with the construction of* J*.*

5 Finding an optimal support

In this section, we address the outer problem (14) of choosing an optimal support. This combinatorial represents a specific k-median problem and is hence NP-complete [6]. Nevertheless, when not considering a discrepancy but probability metrics $d_{\mathcal{F}_c}$ with ζ-structure defined by (4) and $c(\omega, \tilde{\omega}) = \|\omega - \tilde{\omega}\|$, (14) can be formulated as a mixed-integer linear program that can be solved numerically for moderate values of N and n by available optimization software.
Furthermore, heuristical approaches have been developed for probability metrics with ζ-structure, cf., e.g., [7]. We shortly recall their forward and backward algorithms that have been shown to be fast and to provide often nearly optimal solutions. They determine index subsets $J^{[n]}$ and $J^{[N-n]}$, respectively, of $\{1, \ldots, N\}$. These sets of cardinality n represent the support of the reduced measure $\mathbb{Q}$. Adapted to our framework of discrepancy distances, the algorithms read as follows.

Algorithm 5.1. *Forward selection.*

Step [0]: $J^{[0]} := \emptyset$.

Step [i]: $l_i \in \operatorname{argmin}_{l \notin J^{[i-1]}} \inf_{q \in S_i} \alpha_{\mathcal{B}}(\mathbb{P}, (\{\xi^{l_1}, \ldots, \xi^{l_{i-1}}, \xi^l\}, q))$,

$J^{[i]} := J^{[i-1]} \cup \{l_i\}$.

Step [n+1]: *Minimize* $\alpha_{\mathcal{B}}(\{\mathbb{P}, (\xi^{l_1}, \ldots, \xi^{l_n}\}, q))$ *subject to* $q \in S_n$.

Algorithm 5.2. *Backward reduction.*

Step [0]: $J^{[0]} := \{1, \ldots, N\}$.

Step [i]: $u_i \in \operatorname{argmin}_{u \in J^{[i-1]}} \inf_{q \in S_{N-i}} \alpha_{\mathcal{B}}(\mathbb{P}, (\{\xi^j | j \in J^{[i-1]} \setminus \{u\}\}, q))$,

$J^{[i]} := J^{[i-1]} \setminus \{u_i\}$.

Step [N-n+1]: *Minimize* $\alpha_{\mathcal{B}}(\{\mathbb{P}, (\xi^j : \ j \in J^{[N-n]}\}, q))$ *subject to* $q \in S_n$.

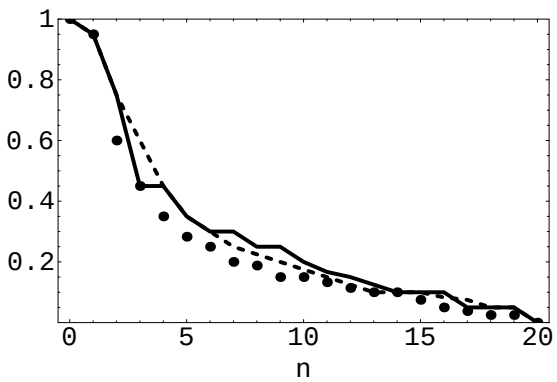

Figure 3: Rectangular discrepancies of Forward selection (solid line), Backward reduction (dashed line) and complete enumeration (dots), depending on the number n of remaining scenarios.

Dealing with discrepancy distances, we meet different problems in solving the combinatorial problem (14). On the one hand, discrepancies do not allow, to the best of our understanding, a reformulation of (14) as a mixed-integer linear program that would be accessible by available solvers. On the other hand, the above-mentioned heuristics do not produce nearly optimal results anymore, cf. Figure 5. This is a consequence of the fact, that the maximum in (6) is attained in many different regions, and, thus, the removal or adding of a single point to the support is often a unfavorable precondition for further reduction or expansion. Furthermore, the backward reduction algorithm becomes significantly slower with increasing N, since Algorithm 4.1 has to determine the optimal redistribution for large values of n, cf. Table 1.

Finally, even for moderate scenario numbers a complete enumeration is very expensive the solution of the inner problem (15) requires higher computational efforts and is not a simple nearest-neighbour projection as in the case of $d_{\mathcal{F}_c}$ with $c(\omega, \tilde{\omega}) = \|\omega - \tilde{\omega}\|$. Running times of complete enumeration can be found in Table 4.

Since sometimes one may be interested in knowing the achievable minimal discrepancy, we suggest the following approach to solve (14) for moderate values of N and to reduce the time needed by complete enumeration. However, the complexity of

(14) quickly increases with the dimension of the problem and real-world stochastic optimization problem are of higher dimension, in general. Consequently, practitioners mostly have to abandon an *optimal* approximation and the following approach is rather of academic interest. For an approach that may be numerically more tractable for larger problems, we refer to Example 6.1, where we adopted a Quasi-Monte Carlo method to tackle the outer problem (14).
The following approach is applicable for both the cell discrepancy studied in [9] and the polyhedral discrepancy. Starting point is the computation of an upper bound $\bar{\alpha}_n$ for the achievable minimal discrepancy by using heuristics, e.g. the forward selection algorithm. Since the optimal discrepancy achievable by m points decreases with increasing m, an optimal tuple of $n < m$ elements may not be contained in any m-tuple with a discrepancy exceeding $\bar{\alpha}_n$. Hence, we can pass through choices u of $m \in \{n+1, \dots, N-1\}$ out of N points to determine some u with a discrepancy exceeding $\bar{\alpha}_n$. Afterwards, to determine the optimal n-tuple, it suffices to evaluate all n-tuples being no subset of any of these u. As soon as we find an n-tuple whose discrepancy falls below the upper bound $\bar{\alpha}_n$, we can update $\bar{\alpha}_n$ and defer the enumeration of $n-$tuples to repeat the iteration of $m-$tuples for $m \in \{n+1, \dots, N-1\}$ to exclude further m- and n-tuples.
Since m-tuples can be seen as admissible solutions to a relaxation of (14), this procedure is close to a branch and bound approach with iterated breadth-first search (Step [2]) and depth-first search (Step [3]). However, a standard branch and bound approach does not perform well since the solution of the redistribution problem along the branch and bound tree is too expensive. We propose the following

Algorithm 5.3. *Branch and bound.*

Step [1]: *Determine an upper bound $\bar{\alpha}_n$ for the minimal discrepancy achievable by a measure supported by n points.*
Set $U = \emptyset$.

Step [2]: *Iterate through <u>some</u> m-tuples w with $m \in \{n+1, \dots, N-1\}$:*
If $w \not\subset u$ for all $u \in U$: calculate optimal redistribution given the support w and the resulting discrepancy. If the latter exceeds $\bar{\alpha}_n$: Add w to U.

Step [3]: *Iterate through all (remaining) n-tuples w.*
If $w \not\subset u$ for all $u \in U$: calculate optimal redistribution on w and the resulting discrepancy. If the latter falls below $\bar{\alpha}_n$, update $\bar{\alpha}_n$ and go to Step [2].

The choice of the tested m-tuples within Step [2] is crucial for the performance of the algorithm. In the remaining part of this section we address this question and suggest a heuristic for this breadth-first search.
The following considerations should be taken into account. On the one hand, the

number of $n-$tuples excluded by an m-tuple with a discrepancy exceeding $\bar{\alpha}_n$ increases with increasing m. On the other hand, with decreasing m, it becomes more likely to find such m-tuples. However, the evaluation of *all* m-tuples becomes quickly too time-consuming when m approaches $N/2$.
Thus, we suggest the following approach for Step [2]. Once having determined the set U from the evaluation of some $N-1,\dots,i+1$-tuples, we can calculate the number of remaining n- and i-tuples, respectively. This is shown by Lemma 5.4 below. The time needed for the evaluation of a single n- or i-tuple, i.e. the costs of the inner problem (15), can be (roughly) estimated to be proportional to the number of potential supporting polyhedra or cells, i.e. in the case of the cell discrepancy $\binom{n+s}{s}$ or $\binom{i+s}{s}$, respectively. We denote the time needed for evaluation of all remaining n- and i-tuples by τ_n^U and τ_i^U. If

$$\tau_i^U \leq \tau_n^U, \tag{34}$$

we invest a certain part $\lambda \in (0,1)$ of the time τ_n^U in the evaluation of some i-tuples, i.e. we evaluate a fraction κ of all remaining i-tuples such that

$$\kappa \cdot \tau_i^U = \lambda \cdot \tau_n^U.$$

This evaluation entails a set $U^\kappa \supset U$. We decide to evaluate *all* remaining i-tuples if and only if

$$\min\left\{\frac{1}{\kappa}(\tau_n^U - \tau_n^{U^\kappa}), \tau_n^U\right\} \geq \tau_i^U. \tag{35}$$

The right-hand side represents the costs of testing all remaining i-tuples, the left-hand side can be interpreted as an extrapolation of the benefit of such a test. Using (34) and the definition of κ, (35) can be written as

$$\tau_n^{U^\kappa} \leq (1-\lambda)\tau_n^U. \tag{36}$$

To this end, we have to calculate the number of remaining n- and i-tuples, given a set U of excluding supersets. This can be done by the following formula that is based on the inclusion-exclusion principle and can be easily proven by induction over m.

Lemma 5.4. *Consider m finite sets $u_1,\dots,u_m$ and $n \in \mathbb{N}, n < \#u_i,\ i = 1,\dots,m$. The number of sets of cardinality n being no subset of any u_i, $i \in \{1,\dots,m\}$, is given by*

$$\binom{N}{n} - \sum_{k=1}^{m}(-1)^{k+1} \sum_{\substack{I\subset\{1,\dots,m\}\\ \#I=k}} \mathbb{1}_{\{|\cap_{i\in I}u_i|\geq n\}} \cdot \binom{|\cap_{i\in I}u_i|}{n}. \tag{37}$$

Remark 5.5. *Since the evaluation of* (37) *requires the determination of 2^m-1 intersections, one could think about using an estimation of (37) that is cheaper to evaluate. Indeed, given an even $\underline{m} < m$, (37) can be bounded from below and above*

by taking the first sum only over $k \leq \underline{m}$ and $k \leq \underline{m}+1$, respectively. However, these so-called Bonferroni inequalities, cf., eg., [5], do not entail a useful estimate of (37) since the sums are strongly fluctuating in $\underline{m}$. Furthermore, such estimates do not lead to a significant speed-up, in general, because the condition $\mathbb{1}_{\{|\cap_{i\in I} u_i| \geq n\}}$ allows to abort the computation in many cases, anyway.
On the other hand, numerical experiences with substituting $\mathbb{1}_{\{|\cap_{i\in I} u_i| \geq n\}}$ by $\mathbb{1}_{\{|\cap_{i\in I} u_i| \geq n+j\}}$ were encouraging for small values of j. However, we do not pursue this approach further on and evaluate (37) *only if m is smaller than a certain threshold ϑ.*

We propose the following detailed form for Step [2] of Algorithm 5.3:

Algorithm 5.6. *Breadth-first search heuristics.*

Step [2a]:	*Set $i := \bar{L} \in \{n+1, \ldots, N-1\}$ and $U := \emptyset$.*
Step [2b]:	*Set $i := i-1$. If $i = n$, proceed with Step [3].*
Step [2c]:	*Go through all already evaluated i-tuples and compare their saved discrepancies with $\bar{\alpha}_n$. Add tuples with a discrepancy exceeding $\bar{\alpha}_n$ to U.*
Step [2d]:	*If* all *i-tuples have been already evaluated go to [2b].*
Step [2e]:	*If $\lvert U\rvert \leq \vartheta$ calculate τ_i^U and τ_n^U.*
Step [2f]:	*Evaluate a fraction $\kappa = \lambda \tau_n^U / \tau_i^U$ of all i-tuples, save their discrepancies and determine U^κ. Update $U := U^\kappa$.*
Step [2g]:	*If $\tau_i^U > \tau_n^U$ or $\kappa \geq 1$ go to [2b].*
Step [2h]:	*If $\lvert U^\kappa\rvert \leq \vartheta$ calculate $\tau_n^{U^\kappa}$ and check whether (36) is satisfied. If this is the case, go to [2j].*
Step [2i]:	*If $\lvert U^\kappa\rvert > \vartheta$ and $\tau_i^U < \sigma \cdot \tau_n^U$ go to [2j], else proceed with [2b].*
Step [2j]:	*Evaluate all i-tuples, save their discrepancies, and update U. Go to [2b].*

Remark 5.7. *Since the time needed for evaluation of a $m-$tuple increases with m, it is reasonable to use $\bar{L} \in \{n+1, \ldots, N-1\}$ in Step [2a] whenever $n \ll N$.*
The vast majority of the computational time is needed for the computation of discrepancies. Thus, all computed discrepancies are saved and again taken into account in Step [2c] whenever the upper bound $\bar{\alpha}_n$ decreases.
In Step [2f], a sample of size κ is taken from the i-tuples. Thereby, κ is updated in terms of the τ_j^U as long as this does not take too much time, i.e. whenever $\lvert U\rvert \leq \vartheta$. This is verified in Step [2e].
Whenever $\kappa \geq 1$ in [2g], all $i-$tuples have been evaluated in [2f], thus we can proceed with [2b] and $i-1$.

When the evaluation of the κ-sample entails $|U^\kappa| > \vartheta$ for the first time, we do not estimate the worth of an evaluation of all i-tuples via (36). Instead of that, we compare τ_i^U with τ_n^U and decide to compute all $i-$tuples if this seems to be comparatively cheap. This is done in Step [2i].

Table 1: Running times (in seconds) of Algorithm 4.1 for different problem parameters.

		k	n=5	n=10	n=15	n=20
		3	0.01	0.04	0.56	6.02
N=100	$\mathbb{R}^3$	6	0.03	1.03	14.18	157.51
		9	0.15	7.36	94.49	948.17
		4	0.01	0.19	1.83	17.22
N=100	$\mathbb{R}^4$	8	0.11	5.66	59.28	521.31
		12	0.67	39.86	374.15	3509.34
		3	0.01	0.05	0.53	4.28
N=200	$\mathbb{R}^3$	6	0.03	0.76	11.80	132.21
		9	0.12	4.22	78.49	815.79
		4	0.01	0.20	2.56	41.73
N=200	$\mathbb{R}^4$	8	0.11	4.44	73.70	1042.78
		12	0.74	28.29	473.72	6337.68
		3	0.01	0.05	0.37	1.79
N=300	$\mathbb{R}^3$	6	0.04	0.75	8.35	42.44
		9	0.16	4.39	52.47	259.44
		4	0.03	0.22	2.83	61.21
N=300	$\mathbb{R}^4$	8	0.13	6.29	94.87	2114.38
		12	0.63	42.03	622.12	11666.28

6 Numerical results

All algorithms have been implemented in C++, the halfspace representation of $\operatorname{pos} W$ has been determined by Fukuda's *cdd+* ([4]), and the linear program (19) has been solved with CPLEX 10.0 ([12]). The following numerical results have been realized on a Pentium 4 with 3 GHz CPU and 1 GB RAM.

Table 1 shows running times of Algorithm 4.1 for the optimal redistribution given a fixed support for different problem parameters. In $\mathbb{R}^s$, the case $k = s$ stands for the rectangular discrepancy. Running time quickly increases with increasing support size n and number of facets k, due to the fact that the number of potential supporting polyhedra is equal to $\binom{n+1}{2}^k$. For $n = 5, k = 3$ and $n = 20, k = 12$ the

Table 2: Polyhedral and rectangular discrepancies for different problem parameters.

		k	n=5	n=10	n=15	n=20
N=100	$\mathbb{R}^3$	3	0.66	0.48	0.41	0.36
		6	0.71	0.48	0.42	0.36
		9	0.71	0.48	0.42	0.39
N=100	$\mathbb{R}^4$	4	0.85	0.53	0.38	0.31
		8	0.86	0.53	0.38	0.31
		12	0.86	0.53	0.38	0.31
N=200	$\mathbb{R}^3$	3	0.54	0.47	0.35	0.32
		6	0.56	0.47	0.37	0.34
		9	0.56	0.48	0.37	0.34
N=200	$\mathbb{R}^4$	4	0.80	0.55	0.46	0.40
		8	0.80	0.56	0.50	0.46
		12	0.80	0.56	0.50	0.46
N=300	$\mathbb{R}^3$	3	0.72	0.48	0.37	0.29
		6	0.78	0.52	0.42	0.31
		9	0.78	0.52	0.42	0.31
N=300	$\mathbb{R}^4$	4	0.86	0.68	0.51	0.44
		8	0.86	0.68	0.51	0.46
		12	0.86	0.68	0.51	0.46

latter is equal to 3375 and 7.36×10^{27}, respectively. The dependency of the running time in terms of the initial number of scenarios N appears to be linear. Table 2 shows the resulting polyhedral and rectangular discrepancies. As one would expect, these are increasing in k and decreasing in n.

Table 3 illustrates the increase of Forward Selection Algorithm 5.1's running time, when reducing initial measures supported by $N = 100$ atoms in $\mathbb{R}^3$ and $\mathbb{R}^4$ w.r.t. the rectangular discrepancy. The growth of the running time is due to the fact that the inner problem, which becomes more complex with increasing dimension and number of remaining scenarios, has to be solved very often in the course of a Forward Selection. Consequently, this heuristic seems to be not appropriate for higher dimensional problems. We refer again to Example 6.1, where another heuristic is used to tackle the outer problem (14).

Table 4 compares the results of a complete enumeration, the branch and bound Algorithm 5.6, and the Forward Selection Algorithm 5.1 in the case of the cell discrepancy. The initial measures were supported by $N = 20$ atoms in $\mathbb{R}^2$ and $\mathbb{R}^4$, respectively. The percentage values in brackets specify the relative excess of the minimal discrepancy achieved by Forward Selection over the optimal value. The parameters used for the breadth-first search are $\bar{L} = \min\{N - 1, n + 4\}$, $\lambda = 0.01$,

Table 3: Growth of running times (in seconds) of Forward Selection Algorithm 5.1 for the rectangular discrepancy and different problem parameters.

N=100	n=5	n=10	n=15
$\mathbb{R}^2$	0.21	2.07	17.46
$\mathbb{R}^3$	0.33	8.40	230.40
$\mathbb{R}^4$	0.61	33.69	1944.94

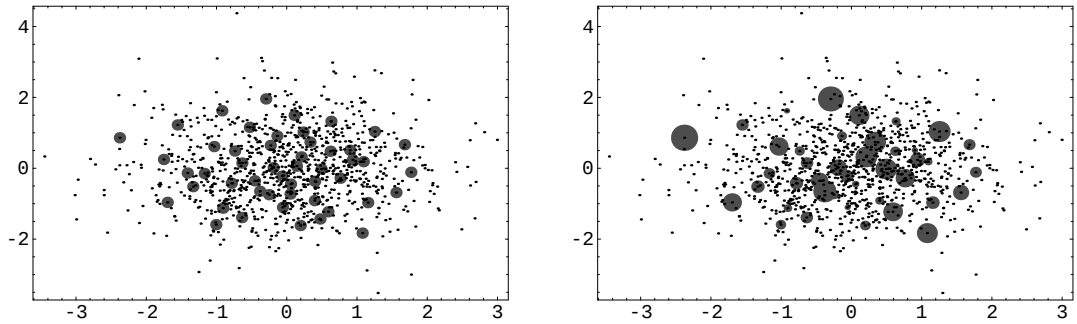

Figure 4: $N = 1,000$ initial scenarios (black points) and $n = 50$ reduced scenarios (gray points) of Example 6.1. Radii of the points are proportional to their probabilities. The left figure shows the uniformly weighted points obtained by Quasi-Monte Carlo, the right figure shows their probabilities readjusted with Algorithm 4.1.

$\sigma = 0.3$ and $\vartheta = 50$; the time needed by enumeration is significantly reduced, by up to 95% ($\mathbb{R}^2$, $n = 11$).

With regard to the complexity of the inner problem (15), we adopted a heuristic for the outer problem (14), within that (15) has to be solved only once. More precisely, to approximate the initial measure of Example 6.1, we used a Quasi-Monte Carlo (QMC) approach, based on the first n points of the Halton sequence with bases 2 and 3, cf. [15]. It is not completely clear, to the best of our knowledge, how such a low discrepancy sequence, initially designated to be close to the uniform distribution on the unit cube, should be tranformed to approximate an *arbitrary* probability distribution. However, there exist approaches for special classes of distributions. For Example 6.1, we applied the method of Hlawka and Mück [11]. The resulting (uniformly weighted) points are shown for $n = 50$ on the left side of Figure 4. The right side of Figure 4 shows the probabilities optimally readjusted w.r.t. the rectangular discrepancy by Algorithm 4.1.

The first plot of Figure 5 shows the rectangular discrepancies between the initial distribution of Example 6.1 and the approximations obtained by QMC and its readjustment, respectively, as well as the corresponding running times (in seconds) of

Table 4: Running times (in seconds) of complete enumeration, the branch and bound Algorithm 5.6, and Forward Selection Algorithm 5.1. The terms in brackets show the gaps between the achievable minimal discrepancies and the ones obtained by the heuristics.

	$\mathbb{R}^2$				$\mathbb{R}^4$			
n	complete enum.	branch & bound	forward selection		complete enum.	branch & bound	forward selection	
1	0.04	0.96	0.03	(0%)	0.03	21.24	0.03	(0%)
2	0.35	1.22	0.06	(43%)	0.38	21.47	0.06	(8%)
3	2.30	2.57	0.10	(27%)	2.44	22.85	0.10	(50%)
4	10.19	6.44	0.14	(75%)	11.58	28.51	0.15	(35%)
5	34.11	14.82	0.18	(0%)	43.61	49.39	0.18	(50%)
6	89.71	28.67	0.22	(0%)	134.64	99.71	0.25	(40%)
7	187.50	34.12	0.24	(20%)	349.39	175.88	0.31	(11%)
8	325.35	44.68	0.28	(25%)	779.07	376.18	0.45	(0%)
9	459.34	57.68	0.31	(0%)	1490.33	385.53	0.61	(14%)
10	541.39	84.12	0.34	(20%)	2260.59	730.89	0.77	(22%)
11	524.13	26.47	0.38	(33%)	3089.74	1698.63	1.04	(9%)
12	419.53	27.76	0.42	(24%)	3462.72	686.11	1.40	(20%)
13	278.62	16.88	0.45	(50%)	3156.71	778.05	1.80	(25%)
14	150.85	12.79	0.48	(50%)	2307.49	1267.31	2.38	(0%)
15	65.80	18.61	0.50	(50%)	1342.47	223.15	3.10	(25%)
16	22.46	19.55	0.52	(25%)	605.27	222.93	3.94	(50%)
17	5.80	1.85	0.56	(33%)	200.04	113.75	4.98	(100%)
18	1.06	1.33	0.57	(0%)	46.73	53.51	6.19	(100%)

Algorithm 4.1. The discrepancy can be reduced by Algorithm 4.1 by up to 50% (Consequently, the reduction of the gap to the *minimal discrepancy* will be still larger.) The following example illustrates that a such an optimal adjustment w.r.t. to the right discrepancy may significantly improve the approximation quality.

Example 6.1. *Consider a random variable* $\boldsymbol{\xi} = (\boldsymbol{\xi}_1, \boldsymbol{\xi}_2)$ *taking values in* $\mathbb{R}^2$*, some* $p \in [0,1]$*,* $L \geq 0$*, and the following chance-constrained optimization problem of type* (10)*:*

$$\textit{minimize } \left\{x_1 + x_2 : x = (x_1, x_2) \in \mathbb{R}^2, \mathbb{P}\left(\boldsymbol{\xi} - x \in [0, L] \times [0, L]\right) \geq p\right\} \tag{38}$$

This is a prototype model for so-called reservoir constraints with upper and lower level restrictions, as they are applied, for instance, in water management [19] or chemical engineering [10].
We assume that $\boldsymbol{\xi}$*'s distribution consists of* $1,000$ *uniformly weighted points, sampled from the standard normal distribution in* $\mathbb{R}^2$*. Due to its simple form and low*

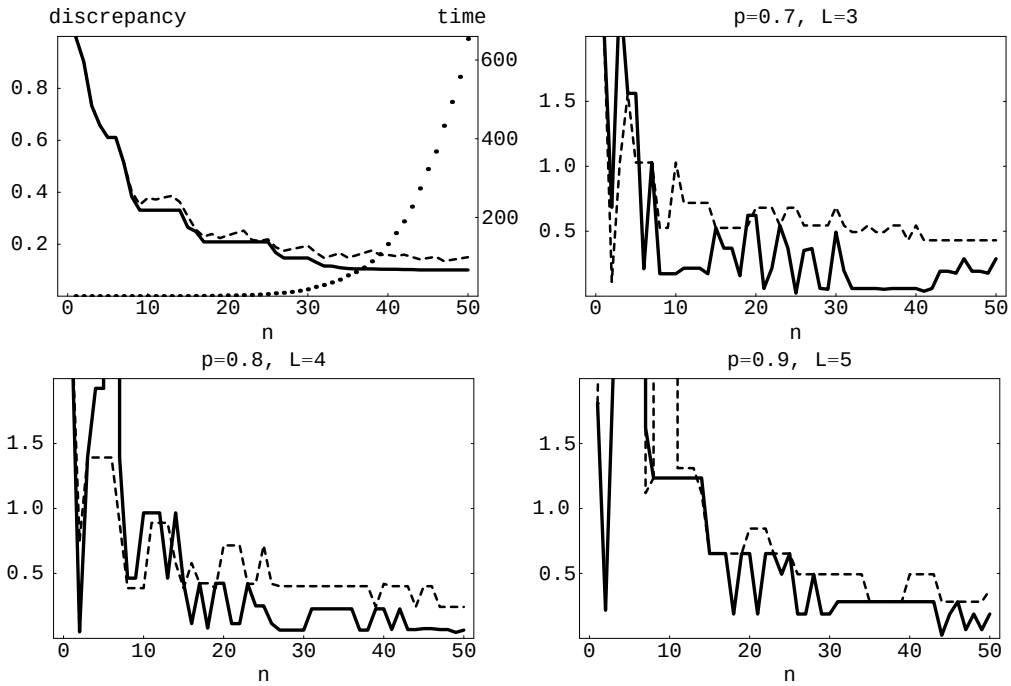

Figure 5: Rectangular discrepancies and deviation of optimal values of Example 6.1 of Quasi-Monte Carlo (dashed line) and QMC readjusted by Algorithm 4.1 (solid line), depending on the number n of remaining scenarios.

dimensionality, problem (38) *can be solved by enumeration. We compared the optimal value with those obtained by the above-mentioned Quasi-Monte Carlo approach and its readjusted modification, supported by up to* $n = 50$ *atoms.*
The plots of Figure 5 show the relative deviation of the optimal values obtained by the approximations from the intial optimal value for different parameters of p *and* L*. It can be seen that a QMC approximation that has been adjusted by a call of Algorithm 4.1 performs in most cases significantly better than its unadjusted counterpart.*

Appendix

In this section, we point out why it seems to be difficult to directly employ the *extended* $\mathcal{B}$*-discrepancy* $\zeta_{r,\mathcal{B}}(\mathbb{P}, \mathbb{Q})$, introduced in Section 1, for scenario reduction.

For the set $\mathcal{I}_{\mathcal{B}}$ of critical index sets $I(B)$, $B \in \mathcal{B}$, defined in Section 2, we obtain for the two (discrete) probability measures $\mathbb{P}$ and $\mathbb{Q}$

$$\mathbb{P} = \sum\nolimits_{i=1}^{N} p_i \delta_{\xi^i} \quad \text{and} \quad \mathbb{Q} = \sum\nolimits_{j=1}^{n} q_j \delta_{\xi^j},$$

that the extended $\mathcal{B}$-discrepancy $\zeta_{r,\mathcal{B}}(\mathbb{P},\mathbb{Q})$ is of the form

$$\begin{aligned}\zeta_{r,\mathcal{B}}(\mathbb{P},\mathbb{Q}) &= \sup\Big\{\sum_{i\in I} p_i f(\xi^i) - \sum_{j\in I\cap\{1,\dots,n\}} q_j f(\xi^j) : f\in\mathcal{F}_r(\Xi), I\in\mathcal{I}_{\mathcal{B}}\Big\}\\ &= \max_{I\in\mathcal{I}_{\mathcal{B}}}\sup_{u\in\mathcal{U}_r}\Big\{\sum_{i\in I} p_i u_i - \sum_{j\in I\cap\{1,\dots,n\}} q_j u_j\Big\},\end{aligned} \tag{39}$$

with $\mathcal{U}_r := \{u=(u_1,\dots,u_N)\in\mathbb{R}^N : |u_i|\le \max\{1,\|\xi^i\|^r\}, u_i-u_j\le c_{ij}, i,j\in\{1,\dots,N\}\}$ and $c_{ij} := c_r(\xi^i,\xi^j) := \max\{1,\|\xi^i\|^{r-1},\|\xi^j\|^{r-1}\}\|\xi^i-\xi^j\|$ for all $i,j\in\{1,\dots,N\}$. Note that (39) corresponds to the identity (18) for the $\mathcal{B}$-discrepancies. Consequently, when dealing with *extended* discrepancies, the inner problem (15) has the form

$$\begin{aligned}&\text{minimize}\quad t\quad \text{subject to}\quad q\in S_n,\\ &\sup_{u\in\mathcal{U}_r}\Big\{\sum_{i\in I} p_i u_i - \sum_{j\in I\cap\{1,\dots,n\}} q_j u_j\Big\}\le t\ \text{ for all } I\in\mathcal{I}_{\mathcal{B}}.\end{aligned} \tag{40}$$

In contrast to (19), the supremum on the left side does not depend monotonously on $|I|$ when $I\cap\{1,\dots,n\}$ and q are fixed, cf. Example A.1. Thus, to the best of our understanding, passing from (40) to the reduced system of critical index sets $\mathcal{I}^*_{\mathcal{B}}$ and to an analogue of problem (22) in Section 2 seems to be impossible.

Example A.1. *We consider $N=4$, $n=1$, $r=1$, and $\xi^i=i$ for $i=1,2,3$, and $\xi^4=1-\varepsilon$, $p_i=.25, i=1,\dots,4$, and $q_1=1$. Let $\mathcal{B}$ denote the system of all closed intervals on $\mathbb{R}$. We consider the critical index set $\{1,2\}$ and two enlarged sets and calculate the corresponding suprema:*

I	$\{1,2\}$	$\{1,2,3\}$	$\{1,2,4\}$
$\sup_{u\in\mathcal{U}_r}\{\dots\}$	0.75	1	$0.5+0.25\varepsilon$

where a tuple $u^*\in\mathcal{U}_r$ realizing the suprema for these index sets is given by

$$(u_1^*,\dots,u_4^*) = (-1,0,1,-1+\varepsilon).$$

It can be seen that $\sup_{u\in\mathcal{U}_r}\{\dots\}$ does not depend monotonously on $|I|$.

References

[1] D. Avis: lrs homepage, 2005 (`http://cgm.cs.mcgill.ca/~avis/C/lrs.html`).

[2] D. Avis and K. Fukuda: A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra, *Discrete and Computational Geometry* 8 (1992), 295–313.

[3] J. Dupačová, N. Gröwe-Kuska, and W. Römisch: Scenario reduction in stochastic programming: An approach using probability metrics, *Mathematical Programming* 95 (2003), 493–511.

[4] K. Fukuda: cdd and cddplus homepage, 2005 (`http://www.ifor.math.ethz.ch/~fukuda/cdd.home/cdd.html`).

[5] J. Galambos and I. Simonelli: *Bonferroni-type Inequalities with Applications*, Springer, New York 1996.

[6] M.R. Garey and D.S. Johnson: *Computers and Intractability - A Guide to the Theory of NP-Completeness*, W.H. Freeman, 1979.

[7] H. Heitsch and W. Römisch: Scenario reduction algorithms in stochastic programming, *Computational Optimization and Applications* 24 (2003), 187–206.

[8] H. Heitsch and W. Römisch: A note on scenario reduction for two-stage stochastic programs, *Operations Research Letters* (2007) (to appear).

[9] R. Henrion, C. Küchler, and W. Römisch: Scenario reduction in stochastic programming with respect to discrepancy distances, Preprint 354, DFG Research Center Matheon *Mathematics for key technologies*, 2006 and *Computational Optimization and Applications* (to appear).

[10] R. Henrion and A. Möller: Optimization of a continuous destillation process under random inflow rate, *Computers & Mathematics with Applications* 45 (2003), 247–262.

[11] E. Hlawka and R. Mück: Über eine Transformation von gleichverteilten Folgen II, *Computing* 9 (1972), 127–138.

[12] ILOG CPLEX 10.0 (`http://www.ilog.com/products/cplex`).

[13] T. S. Motzkin, H. Raiffa, G. L. Thompson, and R. M. Thrall: The double description method, Contrib. Theory of Games, II, Ann. Math. Stud. No. 28, 51-73 (1953).

[14] R. Mück and W. Philipp: Distances of probability measures and uniform distribution mod 1, *Mathematische Zeitschrift* 142 (1975), 195–202.

[15] H. Niederreiter: *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, 1992.

[16] H. Niederreiter and J.M. Wills, Diskrepanz und Distanz von Maßen bezüglich konvexer und Jordanscher Mengen, *Mathematische Zeitschrift* 144 (1975), 125–134.

[17] M. P. Nowak, R. Schultz and M. Westphalen: A stochastic integer programming model for incorporating day-ahead trading of electricity into hydro-thermal unit commitment, *Optimization and Engineering* 6 (2005), 163–176.

[18] R. Nürnberg and W. Römisch: A two-stage planning model for power scheduling in a hydro-thermal system under uncertainty, *Optimization and Engineering* 3 (2002), 355–378.

[19] A. Prékopa and T. Szántai: On optimal regulation of a storage level with application to the water regulation of a lake, *European Journal of Operational Research* 3 (1979), 175–189.

[20] S. T. Rachev: *Probability Metrics and the Stability of Stochastic Models*, Wiley, Chichester 1991.

[21] R. T. Rockafellar and R. J-B Wets: *Variational Analysis*, Springer, Berlin 1998.

[22] W. Römisch: Stability of stochastic programming problems, in *Stochastic Programming*, Handbooks in Operations Research and Management Science Vol. 10, Elsevier, Amsterdam, 2003, 483–554.

[23] W. Römisch and S. Vigerske: Quantitative stability of fully random mixed-integer two-stage stochastic programs, Preprint 367, DFG Research Center Matheon Berlin, 2007 and submitted to Optimization Letters.

[24] A. Ruszczyński and A. Shapiro (eds.): *Stochastic Programming*, Handbooks in Operations Research and Management Science Vol. 10, Elsevier, Amsterdam, 2003.

[25] R. Schultz: Rates of convergence in stochastic programs with complete integer recourse, *SIAM Journal on Optimization* 6 (1996), 1138–1152.

[26] R. Schultz: Stochastic programming with integer variables, *Mathematical Programming* 97 (2003), 285–309.

[27] V. M. Zolotarev: Probability metrics, *Theory of Probability and its Applications* 28 (1983), 278–302.